

QL Today

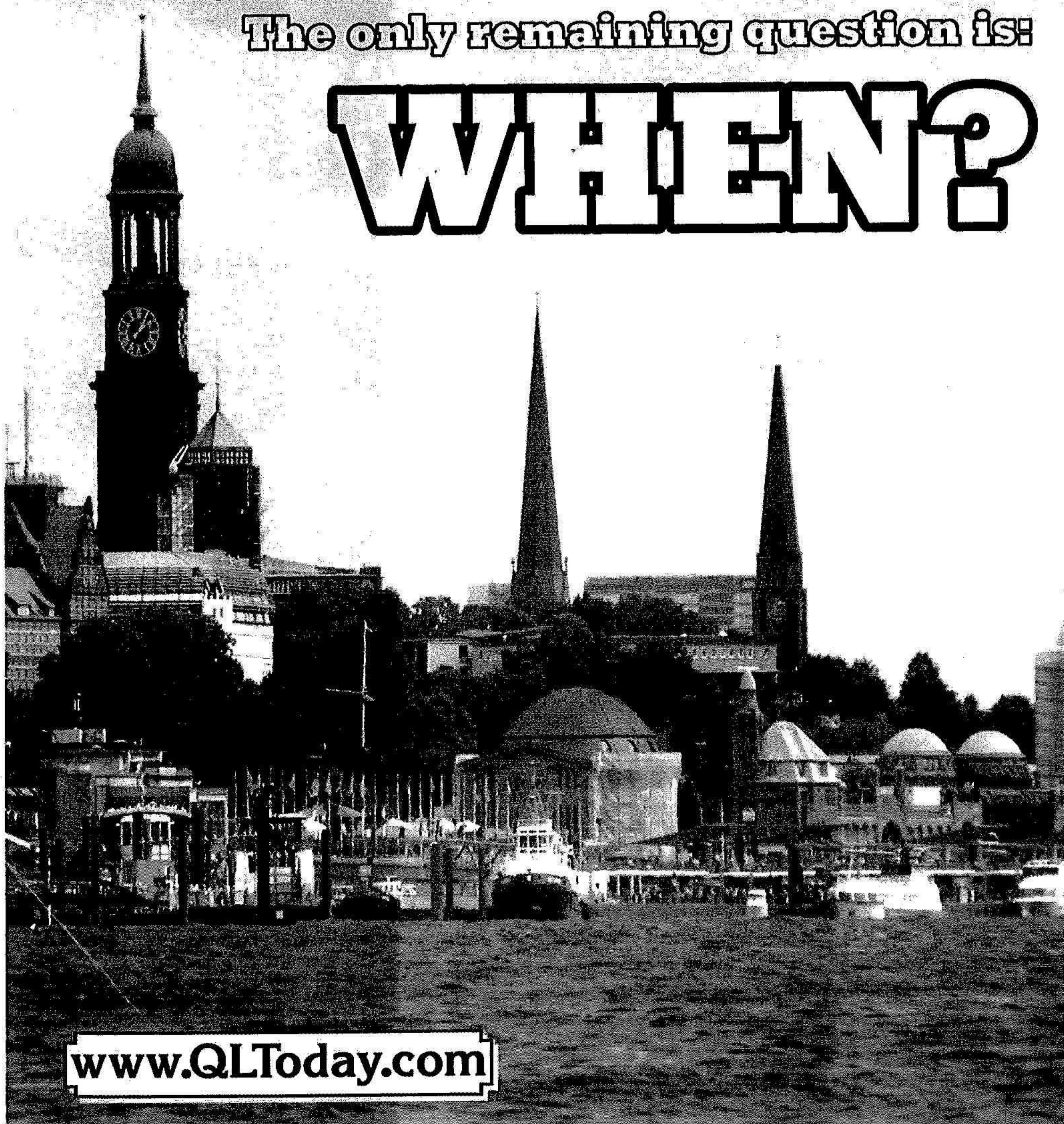
Volume 17
Issue 2
Dec. - Feb.
2012/13

ISSN 1432-5454

The Magazine about QL, QDOS,
Sinclair Computers, SMSQ...

The only remaining question is:

WHEN?



www.QLToday.com

Contents

3	Editorial	
4	News	
6	Review of PS2 Mouse to Games Port Converter	Ian Burkinshaw
10	SQRview	Geoff Wicks
11	The continuing saga of George & Norman	George Gwilt & Norman Dunbar
15	The Games that Time forgot	Peter Scott
16	What went wrong?	Geoff Wicks
20	Machin's Formula for PI	George Gwilt
21	Programming in Assembler, Part 31 LibGen - Library Generator - Part 3	Norman Dunbar
29	I2C Interface for QL Emulators Part 6	Ian Burkinshaw
33	Glossary of Abbreviations and Terms Part 6 - P to Q	Dilwyn Jones & Lee Privett

QL Today

ISSN 1432-5454

German office & Publisher:

Jochen Merz Software	Tel. +49 203 502011
Kaiser-Wilhelm-Str. 302	Fax +49 203 502012
47169 Duisburg	email: smsq@j-m-s.com
Germany	email: QLToday@j-m-s.com

Editor:

Geoff Wicks	Tel. +44 1332 271366
Flat 5b	email: gtwicks@btinternet.com
Wordsworth Avenue	email: QLToday@j-m-s.com
Derby DE24 9HQ	
United Kingdom	

Co-Editor & UK Office:

Bruce Nicholls	Tel +44 20 71930539
38 Derham Gardens	Fax +44 870 0568755
Upminster	email: qltoday@q-v-d.demon.co.uk
Essex RM14 3HA	email: QLToday@j-m-s.com
United Kingdom	

QL Today is published four times a year, our volume begins on beginning of June. Please contact the German or English office for current subscription rates or visit our homepage www.QLTODAY.com.

We welcome your comments, suggestions and articles. YOU make **QL Today** possible. We are constantly changing and adjusting to meet your needs and requirements. Articles for publication should be on a 3.5" disk (DD or HD) or sent via Email. We prefer ASCII, Quill or text87 format. Pictures may be in _SCR format, we can also handle GIF or TIF or JPG. To enhance your article you may wish to include Saved Screen dumps. PLEASE send a hardcopy of all screens to be included. Don't forget to specify where in the text you would like the screen placed.

QL Today reserves the right to publish or not publish any material submitted. Under no circumstances will **QL Today** be held liable for any direct, indirect or consequential damage or loss arising out of the use and/or inability to use any of the material published in **QL Today**. The opinions expressed herein are those of the authors and are not necessarily those of the publisher.

This magazine and all material within is Copyright 2012 Jochen Merz Software unless otherwise stated. Written permission is required from the publisher before the reproduction and distribution of any/all material published herein. All copyrights and trademarks are hereby acknowledged.

If you need more information about the UNZIP program which is used by our BOOT program to unpack the files, we suggest that you visit Dilwyn Jones' web site where you find more information about lots of interesting QDOS software and INFOZIP at <http://www.dilwyn.uk6.net/arch/index.html>

Advertisers

in alphabetical order

Jochen Merz Software (J-M-S)	23
QLForum	13
Quanta	31
QuoVadis Design	19

**The deadline for the next issue
is the 14th of February 2013!**

We were well into the production of this issue when we received news of one of the most significant QL developments in recent years. If we had received it earlier it would have been our lead news story. Wolfgang Lenerz has announced a change to the SMSQ/E licence to give developers greater freedom to use, modify and distribute the source code.

The SMSQ/E licence is 10 years old and has been a source of much controversy in the QL community, particularly in the early years. Although SMSQ/E was a great improvement on QDOS, some QL-ers claimed the restrictive nature of the licence hindered QL development, because it was impossible to port code to and from other computer systems.

These restrictions have now been removed and it will be interesting to see what, if any, impact it has on the nature and range of QL software.

Last issue I looked at the difference between QL and Spectrum users and, in particular, how we have continued to develop the QL. The SMSQ/E licence change should enable us to continue this development.

Although we can be jealous of Spectrum enthusiasts for their greater numbers, there are advantages in being small. There is a long QL history of individuals transforming the QL scene. We are all aware of the contributions of QL-ers like Marcel Kilgus and Nasta, but there are many others as well.

What is universally regarded as the best QL website ever is the initiative of one person. I doubt if anyone has managed to browse the complete content of Dilwyn Jones' website, in itself is a tribute to the vast amount of work of numerous individual QL-ers.

Another project, the preservation of QL games, is also the initiative of one person. That initiative has led, with more than a little help from Quanta, to keyboard membranes still being available to native hardware users. It has also led to a successful commercial website.

Most of these individuals are not remote, anonymous people. Quite often we have met them at shows; heard them give a presentation; read their contributions to forums; seen their articles in QL publications. It is an advantage of being small in numbers.

The UK has always had the largest QL community, although its influence has waned in recent years as major QL developments have moved onto the continent. However, the UK can still boast of being the centre for QL information.

The danger of being the land with the largest QL community is that any decline in numbers can be highly visible and can sap morale. Instead we should be looking at current activity – and there is still plenty – and adapt to being smaller. Why, for example, do we still think of running shows in large halls with lots of tables? And what is wrong with a Quanta subgroup with only three or four members? It is not the numbers that count, but the activities that are taking place.

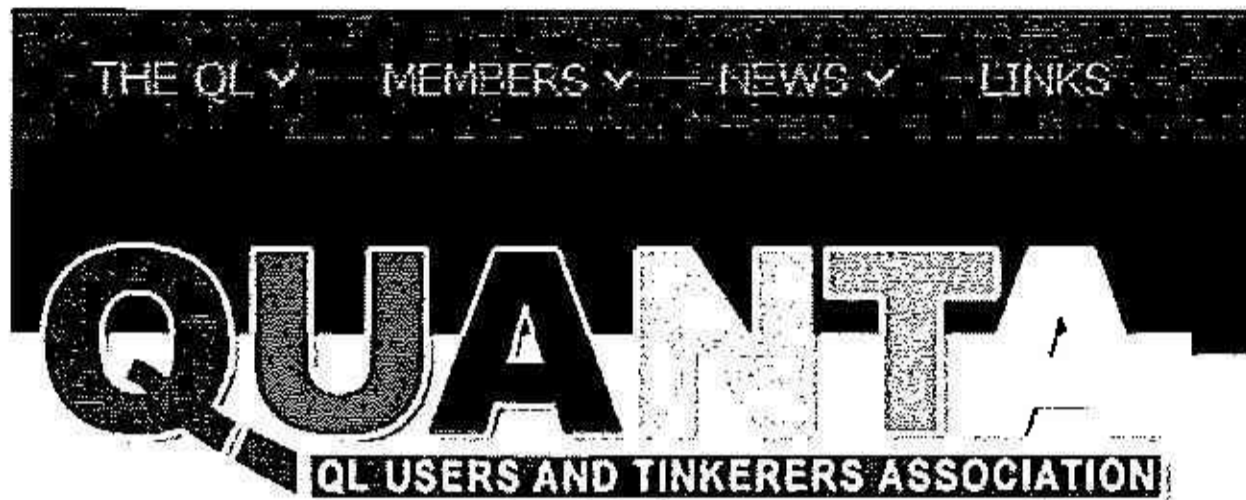
One of the great strengths of the QL is that it can cater well for minority interests. Over the years I have always been amazed by the versatility of QL activity, and the ease of QL programming has encouraged this versatility.

Let us not be frightened of being a small community, but look instead at the strengths that it can and has given us.

QUANTA'S Web Reversal

Quanta has abandoned plans for an ambitious website containing a closed members' area in favour of a simpler information site. In a message on the QL-users email group the webmaster reported:

"the new site will be mainly static so the committee will not be making much effort in updating.."



Quanta has had plans for a website with a closed members' area together with a members' forum for almost 10 years, but has never been able to implement them. The latest efforts began in 2009 when Quanta appointed two webmasters to take on the task. They soon reported that the workload would be too excessive unless work on the site was shared by the whole committee. A solution was sought in a Content Management System (CMS), but this soon ran into technical difficulties. The webmasters then recommended that the Quanta website, then hosted by a free host, would have to be moved to a paid host who would have the technical facilities to support the CMS. The new host made the Quanta website one of the most expensive QL sites to run. In 2011, a year in which Quanta had a £760 deficit, the site cost £346.

In a report to the 2011 Quanta AGM the webmasters justified this expenditure by being able to place a members' forum, a Google map plug in for Quanta events and subgroups, a volunteering module, a mailing list module and interactive photo galleries on the website.

A year later the webmasters reported:

"The website has been pretty static for the last 12 months with very little activity."

No reason for this was given, but in practice only one member of the committee, other than the webmasters, was prepared to use the CMS. Eighteen months after the move to a paid host the only activity on the site was a regularly updated news section, but other content became increasingly out of date.

At a meeting held in mid October, the committee decided to revert to a largely static website using a cheaper host, although the precise meaning of this is not known. Since mid October the Quanta website has been offline on several occasions.

Microvitec Monitor Service Manual

Dilwyn Jones reports:

"Thanks to Norman Dunbar, I have now been able to add a copy of the Microvitec Service Manual for their series-3 colour monitors to the replacement manuals page on my website.

The document is a PDF file, 83 pages long and about three quarters of a megabyte to download from:

<http://www.dilwyn.me.uk/docs/manuals/index.html>

On that page go down to the Monitor Manuals section (about halfway down the page) and you'll find the link just under that for the replacement Cub monitor manual."

JUST WORDS! Dictionary Page

Just Words! has redesigned the dictionaries page on its website. It is now possible to download the files either as dedicated QTYP dictionaries or as text lists.

Geoff Wicks reports that the number of hits he receives on the page is larger than he would expect, and thought that there may be people looking for plain text lists.

www.gwicks.net/justwords.html



DICTIONARIES

JUST WORDS! has a wide range of dictionaries and word lists in QTYP dictionary and simple text formats.

The plain text files will need unzipping, but not the QTYP files. You just need to reverse the letter as a QTYP dictionary.

HELP / ADVICE
DOWNLOAD

JUST WORDS! has made strenuous efforts to check the word lists for accuracy, but please note the larger a word list the more difficult this is to do.

Second Birthday

At the end of November the QL Forum was celebrating its second anniversary. At the end of the first year they had 99 members and the growth has continued in 2011. At the beginning of November there were 146 members with 2,853 posts on 420 different topics.

In the last issue Jochen Merz asked a question about people who were formerly active on the QL scene and what they were now doing.

Someone asked a similar question about Nasta on the QL Forum and received a reply direct from the horse's mouth.

"After moving back to Croatia from the US in 2003 (has it been so many years already?!), I worked for a while as a freelance designer for various small companies, and in early 2005 accepted a job as a full time hardware designer/developer at a small company here in Croatia. Due to a rather complex combination of job and private situations (which could be used to write a book or two if only one could make them sound less bizarre and more believable), my involvement with the QL and Citroen scenes waned to almost nothing. One big problem that developed was that even though I could normally receive and read messages on the QL mailing list, I would not answer - or rather, I could, but the answers, it seems, go to little bit heaven. At the moment the QL list server blocks all ISPs available to me, probably because something en-route blocks a bunch of IP addresses. It's rather odd to see one's name mentioned once in a while and questions asked to which you know the answer, but also being unable to reply."

He added that he had changed jobs:

"As much as I would like to say this change could somehow get me back as an active member in the QL community, I fear I can offer no such guarantee, as the genre of electronics design I am involved with now is quite different. Still, the QL is an 'old love' of sorts and even though the little gray cells are not what they used to be :) there is a lot of information about the QL and it's developments still floating around in there between my ears, and I will be glad to share it as much as available time lets me. Even though my arguably most ambitious project (GoldFire) never materialized, a LOT of thought has gone into that design and a lot of it is still applicable today, so if anyone is interested, feel free to ask."

www.qlforum.co.uk



QL Today Postage Shock

QL Today is to experience a double whammy in the New Year. Not only are Deutsche Post raising their postal rates, but are also rearranging their product types. Among other things the cheap

letter rate for bulk postings that QL Today uses will be abolished. QL Today estimates its postal bill will increase by almost two and a half times.

SMSQ/E LICENCE Changed

At the end of November **Wolfgang Lenerz** made a surprise announcement about changes to the SMSQ/E licence:

"From the SMSQE licence, we will move towards a (new) BSD style licence, which essentially means that anyone can do with the source code anything he likes, and also distribute the compiled code."

The move was greeted with great enthusiasm from several QL-users including some who had requested the change for some time. However **Dave Park** urged a degree of caution:

"This is good news, but let's wait until we see this '(new) BSD-like license' - since it's '(new)' and may not conform to our varied expectations of a standard BSD-like license."

For example, will porting to other architectures be allowed? Will distribution as a firmware image be allowed? Will fees have to be paid?"

In reply **Wolfgang** wrote:

"Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS 'AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."

The Copyright Holders are the two main architects of SMSQ/E, Tony Tebby and Marcel Kilgus, and 10 others who have contributed to its continued development.

DILWYN JONES' Programs

Dilwyn Jones has announced a number of additions to his website:

"A free game and mandelbrot graphics tool have been added to my website this week, along with an update to PCB Design and copies of the Italian versions of the Psion programs for the QL. The game is Rockfall, a colourful Boulderdash clone from the late 1980s. Originally published by CGH Services, the author Andy Toone gave Rich Mellor his blessing to release it freely. It's available to download as a conventional zip file plus, thanks to Rich, a zipped version specially for use in QemuLator. After downloading the QemuLator version, just attach it to one of QemuLator's drive slots and run the game straight from the archive.

Rockfall is available to download from my games page at

<http://www.dilwyn.me.uk/games/index.html>

The Mandelbrot program is called Mandel-Speed, from the same author. The program is available to download in the two versions described above from the graphics page on my website at

<http://www.dilwyn.me.uk/graphics/index.html>

And thanks to a QL user in Italy, I have also been able to make available Italian versions of Quill, Archive, Abacus and Easel, version 2.23 from

<http://www.dilwyn.me.uk/psions/index.html>

And finally, the latest version of Malcolm Lear's PCB Design package, version 7.25, is available to download from the graphics page at

<http://www.dilwyn.me.uk/graphics/index.html>

News

Review of PS2 Mouse to Games Port Converter

by Ian Burkinshaw

As you may have seen, a developer in Italy has developed a converter to convert a PS2 mouse to work either of the games ports on a Black Box QL.

This has all sorts of possibilities but also limitations. Those of you with ICE software can use this interface. As far as I know this software is no longer sold and is not in the public domain. Maybe someone can rescue it and put into the public domain. Also, unless someone has the skills to change it, this interface will not work with PRT_GEN and WMAN which means standard PTR programs will not work. This converter is not a replacement for the QIMI interface, but you can write your own SuperBasic applications, with or without SMSQ, because you can use such functions as KEYROW to read the mouse. Then there is no need to load PRT_GEN and WMAN extensions.

So how does it work? The converter translates the movements of a serial mouse to up, down, left and right keys as used on the games ports of a the QL. It also provides one button (fire) using either of the mouse buttons on a standard two button mouse. The converter is based on a

Microchip PIC16F84A microcontroller. The PIC deals with the serial data stream from the mouse and converts this into the games keys. These keys are represented by what are called by bilateral switches (CMOS IC 4066 or 4016), which is an electronic switch with a control input. High is closed, low is open. So why do we need to do this? Because the QL uses a scanning keyboard. Why scanning keyboards? Having two wires for each key on a keyboard is not very efficient way of working. So the wiring can be simplified by arranging the wiring into row and columns as shown at the top of the opposite page (figure 1). This is in fact the QL keyboard layout.

At each crossing of a row and column there is a switch, which is a key of the keyboard. As shown here in figure 2:

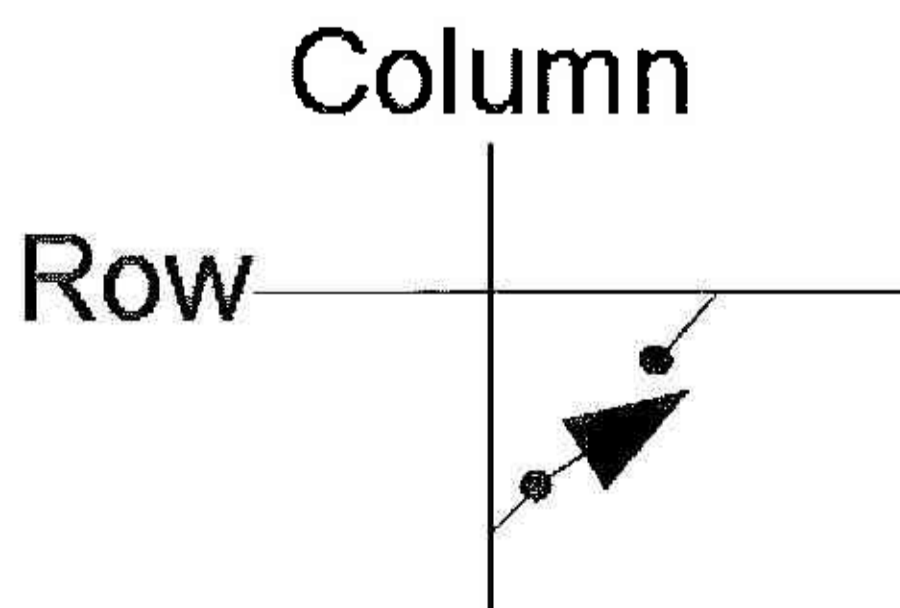


Figure 2

News

6

Figure 1 - QL Keyboard Matrix

J12

	1	2	3	4	5	6	7	8	9	10	11
9	CTRL	SHIFT									ALT
8			+	ESC	+	BAR	4	+	ENTER		
7				X	V	N		<		? /	
6			Z	C	B	M	>	;	}	~	
5			CAPS LOCK	S	F	G	K	:	{	=	
4			!	#	A	D	H	J	L	P	
3			W	TAB	R	Y	I	(O	-	
2			2	Q	E	T	^	U	*)	0
1			F1	F2	F3	\$	%	&	F4	F5	

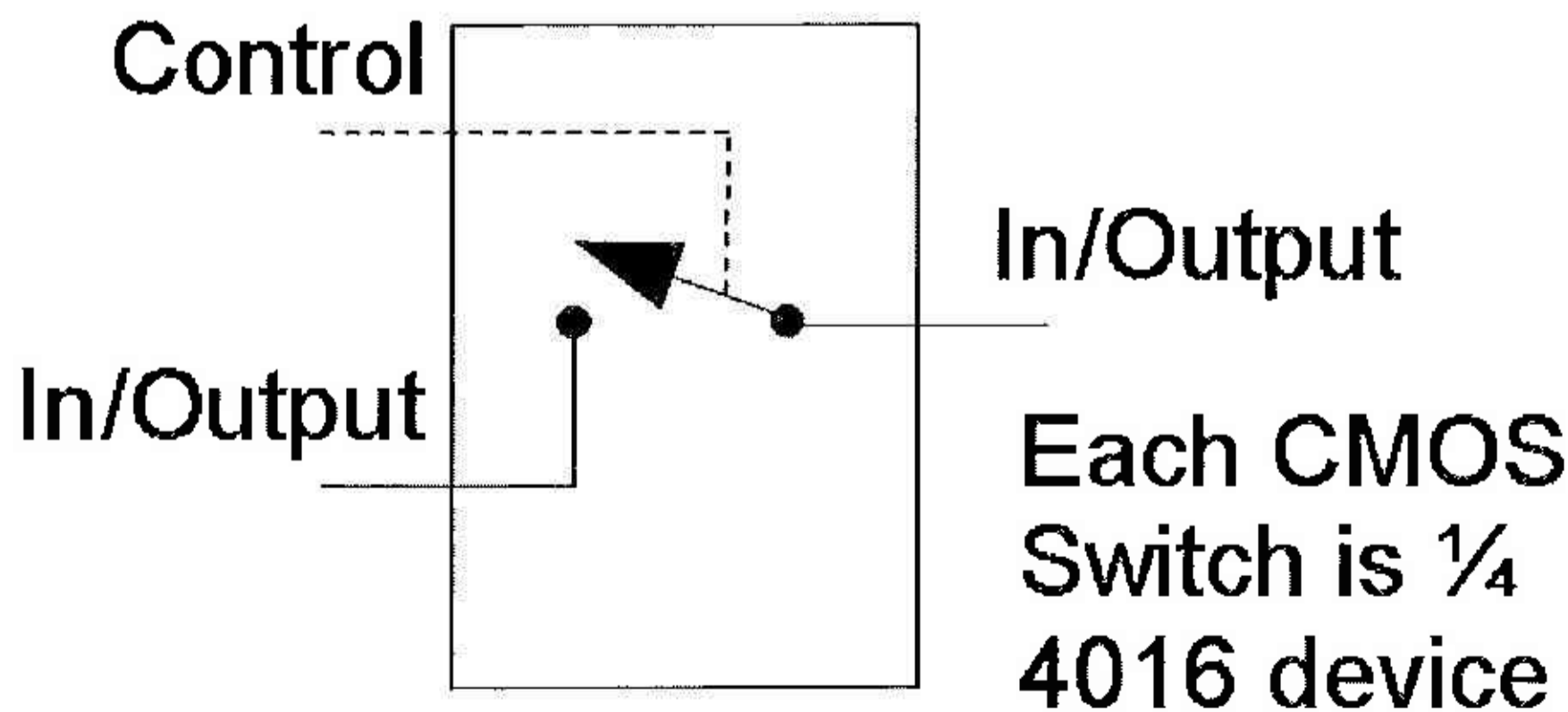
J11

As you can see there are 65 keys on a QL keyboard, which would need at least 66 wires, one for each key plus say ground, however by using the scanning keyboard with row and columns only 20 wires are required.

So how does this work? The columns are fed with a pulse, sequentially, only one column is pulsed at a time. As a key/switch is closed that

pulse is sent to a row. So with a given pulse to a column and the key/switch closed to a given row selects the key required.

Thus what is needed is to replicate the keys/switches with electronics switches. This can be done by using, CD4016 (or 4066) quad bi-lateral switch integrated circuit. A representation of this device is shown below in figure 3.



Two 4016's (or 4066) are used, since each 4016 (or 4066) contains four switches. We use one IC for right, left, up and down. The second IC just uses one switch for the fire button.

There are two game ports on the QL. One is called CTL1 and the other CTL2. The keys are represented as follows (table 1 on the next page):

Figure 3



	CTL1	CTL2
Up	Cursor Up	F4
Down	Cursor Down	F2
Left	Cursor Left	F1
Right	Cursor Right	F3
Fire	Space	F5

Table 1 - QL game ports

If you look at the keyboard diagram on the previous page, you will see that row 8 is common for CTL 1 and row 1 is common for CTL2 so only the columns need to be selected.

Back to reviewing the interface now I have explained how it works.

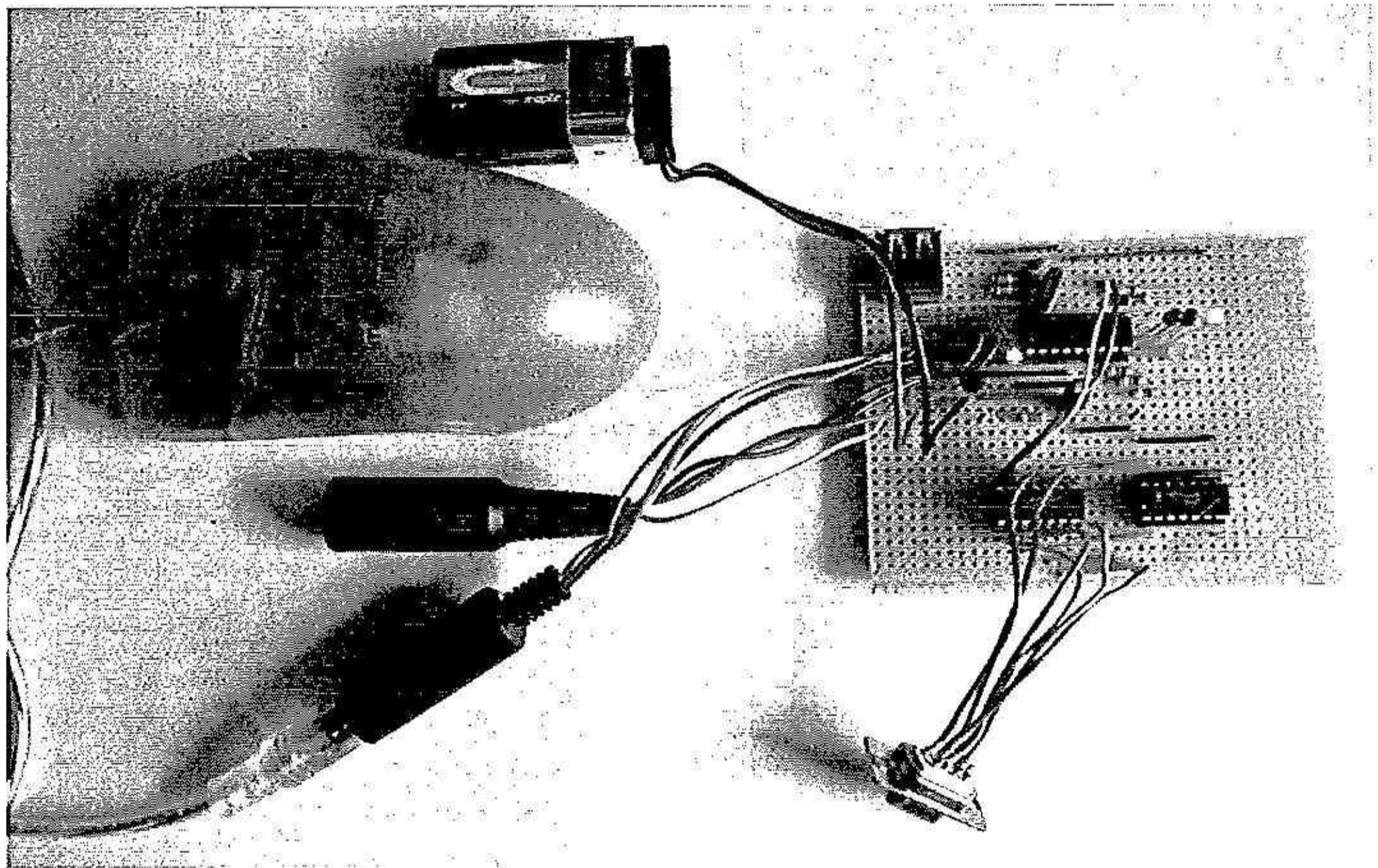
So to start with, you need to download the zip file

<http://www.qlforum.co.uk/viewtopic.php?f=2&t=379>

from the QLForum page. When you open this zip file you get the following files:

GC_QLMouse - Assy.pdf	Component placement PCB looking from component side.
GC_QLMouse - Components side.pdf	As above without PCB tracks
GC_QLMouse - Firmware.HEX	PIC Firmware
GC_QLMouse - Lead..pdf	Shows wiring from the telco connector to 9 pin d-type.
GC_QLMouse - Parts List.pdf	As it says Parts List
GC_QLMouse - PCB.pdf	Again as it says the PCB track diagram

There is enough information here to manufacture the interface. However you may not want to go to the expense of the PCB (printed circuit board). It is true the PCB makes a neater job, but unless you have experience and the equipment, it is not always the easiest to do. However not all is lost since the circuit is not too complicated. I produced mine on a piece of strip board. I have not reproduced the circuit here since it is available from the above zip file.



I added LED's with limiting resistors to the switch (IC 4066) control inputs, just to show me that the converter was working correctly. You do not need to add these if you don't want to. If you look at the Italian pictures on the blog page you will see the developer obtained power via the ROM port. This is OK if you have a spare ROM card. I did not, so I used a 78L05 voltage regulator as shown in the original schematic to provide the 5V required from a 9V battery source. You can make whatever power arrangements you like, the converter just needs 5V. The interface

takes about 15mA, (without the LED's. so with a R22 (PP3) battery it will run the interface for hours. I used rechargeable batteries with no problem.

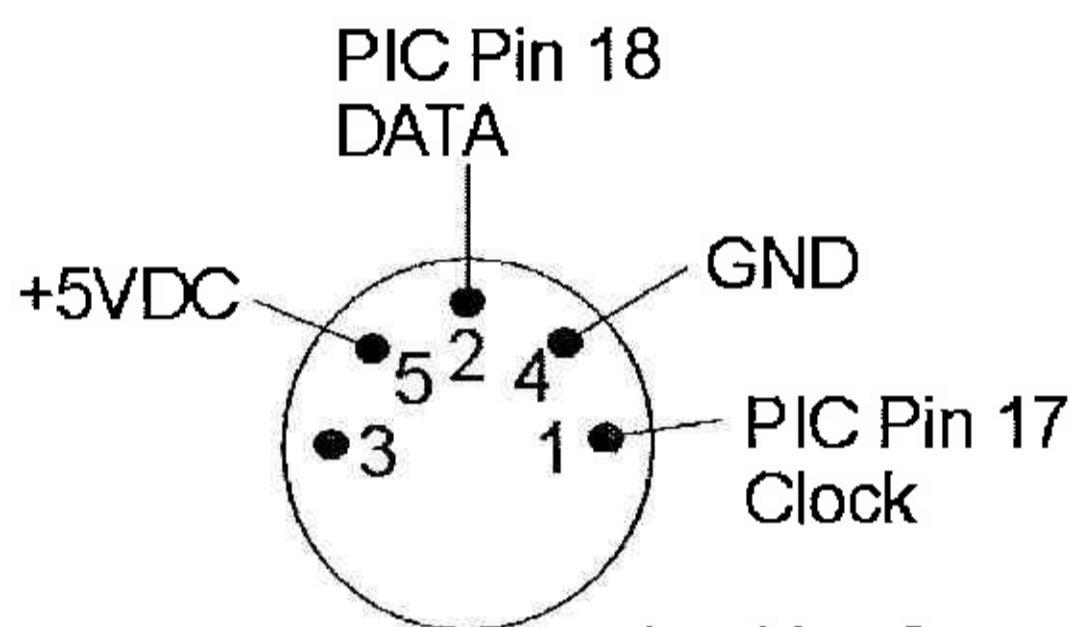
You will need some way to program the PIC. There are low cost PIC programmers around, either on the web or available as kits, from sources such as Maplin's. Of course this adds to the overall cost of the project. I will be looking into this further in the next article of my series on the I2C interface. Since my next project also includes a PIC, so this may help to spread the cost of the

PIC programmer. If you are interested in these projects and you cannot wait, the Vellaman PIC programmer K8048 will support both this project and my next I2C project. It is available from Maplin as a kit. Whatever you do, do make sure the programmer you use supports, in this case, the PIC16F84A chip. Not all do.

Now the mice themselves. You will see from the PCB layout and the circuit diagram that the developers have used a USB connector. There are USB mice, which support the PS2 serial interface. However most USB mice do NOT support PS2. In fact, I could not find a USB mouse with PS2 from any of the major UK suppliers. That is not to say there are none out there, but I just could not find one. However all is not lost. I am

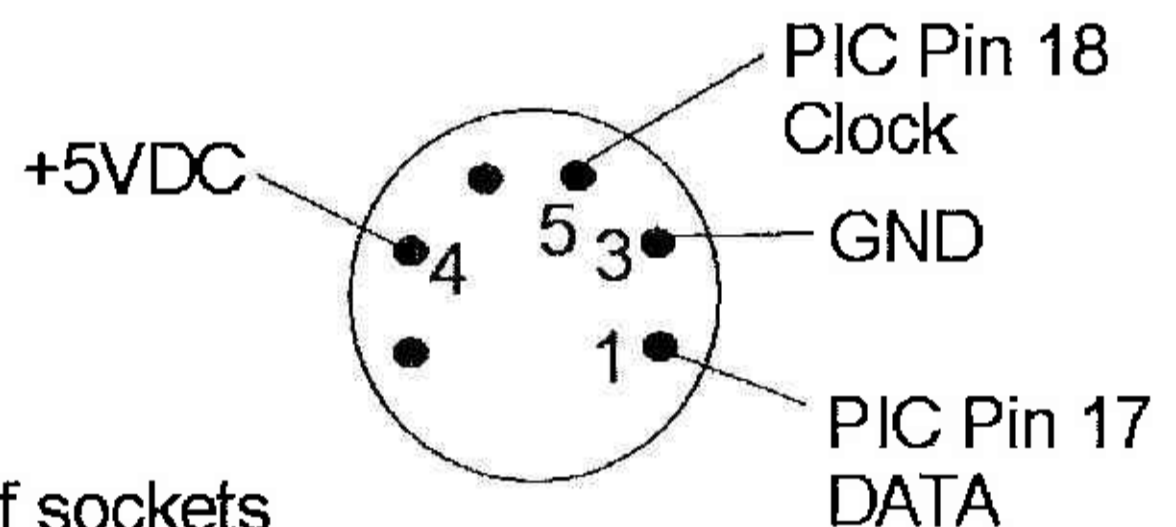
sure most have mice with either a 5 pin DIN or 6 pin Mini DIN connector lying around somewhere. You might find them at car boot or surplus sales. The connections for these types of mice are shown below. You will also find mice with 9 pin D-Type connectors which state they are serial. They can be, but not to the PS2 standard. They use an RS232 serial protocol at 1200 baud. There was software to do this from Albin Hessler, (my copy is dated 1992) called SER Mouse for use with these types of mice. Again as far as I know this is not in the public domain. Note: the Qimi mouse interface used a 9 pin D-Type connector as well, but the Qimi interface does not support serial mice, but used mice that were directly connected to the opto devices and switches inside the mice. This can get a little confusing.

5 Pin Standard DIN Socket (Female)



Looking from rear of sockets

6 Pin Mini DIN Socket (Female)



The BT telephone style connectors are a problem. I have looked around and they don't seem to be available anywhere. I checked with Adrian Ives of Memory Lane Computing and SerUSB fame, since he was the last person to sell any add on that used this connector. He confirmed they are no longer manufactured. He purchased the last from an on-line source, and even these have now gone. Unless, of course, anybody, knows different. There are two types of this connector used on the QL, 603A Right Hand BICC BT type for the RS232 serial ports and 603A Left Hand BICC BT type for the games ports. The plugs for these are called 631W. Current standard BT (631A) connectors may look the same, but in detail they are different and just do not fit. So on my Black Box QL's I have fitted 9 pin D-Types on a flying leads, and soldered to the underside of the QL PCB. So in effect I have made my QL's as if they were the US version, which were fitted with D-Type connectors. This does require the QL to be opened up, so great care is required. If you are not sure 100%, don't do

it. I take no responsibility if it goes wrong. So why did Sinclair not do this on the UK versions - simple, cost.

The cost of this project, should be no more than £15 (without the LED's) to build.

Conclusion, does this interface work? The simple answer is that it does. I built mine up on strip-board in about an hour or so, programmed the PIC and it worked first time without problems. The mouse response was very good, no confusion as to direction at all. There are limitations compared with say the SER Mouse software solution or QIMI hardware interface. For example this interface only supports one mouse button (fire). Other than some games that used the games ports and the ICE software, there is not much ready to run software around that will work with this interface. It could be of use for none PTR developments. Of course for games players, you could build two interfaces with two mice, one for each (CTL) games port on your QL. Now you cannot do that with a PC !!!!

SQRview

by Geoff Wicks

The timing was fortuitous. Last summer I was testing a JPG to PIC transfer program that I hope eventually to post on the Just Words! website. It has been through a number of versions as I passed on suggestions to the author to improve its ease of use and versatility. I needed a simple PIC viewer and almost on cue Bob Spelten delivered the goods with a new program SQRview.

One of the problems with JPG images is that they are usually taken with a digital camera and are a giant size. My camera produces images of about 4Mb. Even a much reduced image would test a PIC viewing program to the limits, but SQRview coped well. For the purpose of this review I converted the JPG images to BMP, which SQRview recognises, and reduced their size to simplify matters at both the PC and QL ends.

whether you want to load as a PIC or a sprite. If your image is smaller than the SQRview default screen then it fits within this window (Fig. 2). If it is larger than the default screen then the window automatically adjusts to the size of the image. To illustrate this I have deliberately chosen an image with unusual proportions (Fig. 3).

My digital camera takes images of 3872 x 2592 pixels and provided I had sufficient memory on my machine SQRview would still have loaded them, but then with vertical and horizontal scrolling bars.

A right click (DO) on the image produces a pop up box giving the properties of the image and giving the option to save or quit.

In his manual Bob Spelten describes the genesis of the program:

"SQRview started life as a test program after a discussion on the QL-users list in June 2009 on 'how to make a scrollable menu for QL screens with EasyPTR'. Then it got a bit out of hand."

It is a situation that many QL hardware and software developers will recognise. It is this constant eagerness to explore and accept new challenges that has made the QL what it is today.

We should be very grateful that it did get out of hand as Bob Spelten has produced a powerful tool for programmers who are eager to use sprites. I have to confess that I have not explored this aspect of the program in any great detail, because my main interest was in using it as a quick viewer.

If you want to convert an image to a sprite you have a number of options to choose. First you can choose between mode 64 and mode 16 sprites. Figures 4 and 5 show the quality difference. You then have various transparency options.

The more you look into this program, the more complicated it becomes, because the options you are given at any one time can vary according to the size of a file, its format and how you want to save it. Recognised file formats include the QLs PIC, PSA, SCR, SPR and PCs BMP and ICO. There are also

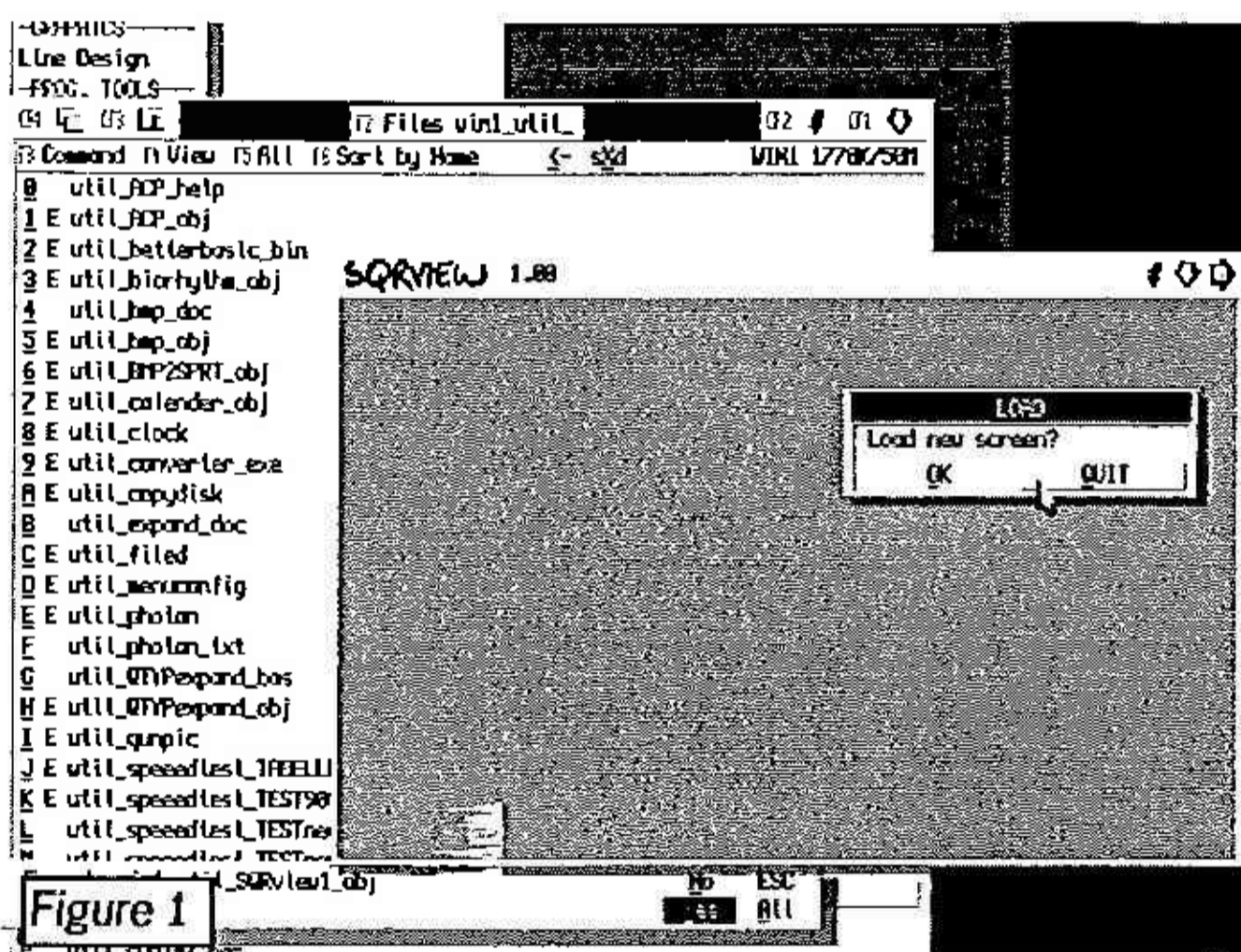


Figure 1

Loading is easy. When you start the program there is a simple menu (Fig. 1) and clicking on load brings up the menu extensions to help you locate the image. You are asked a question on

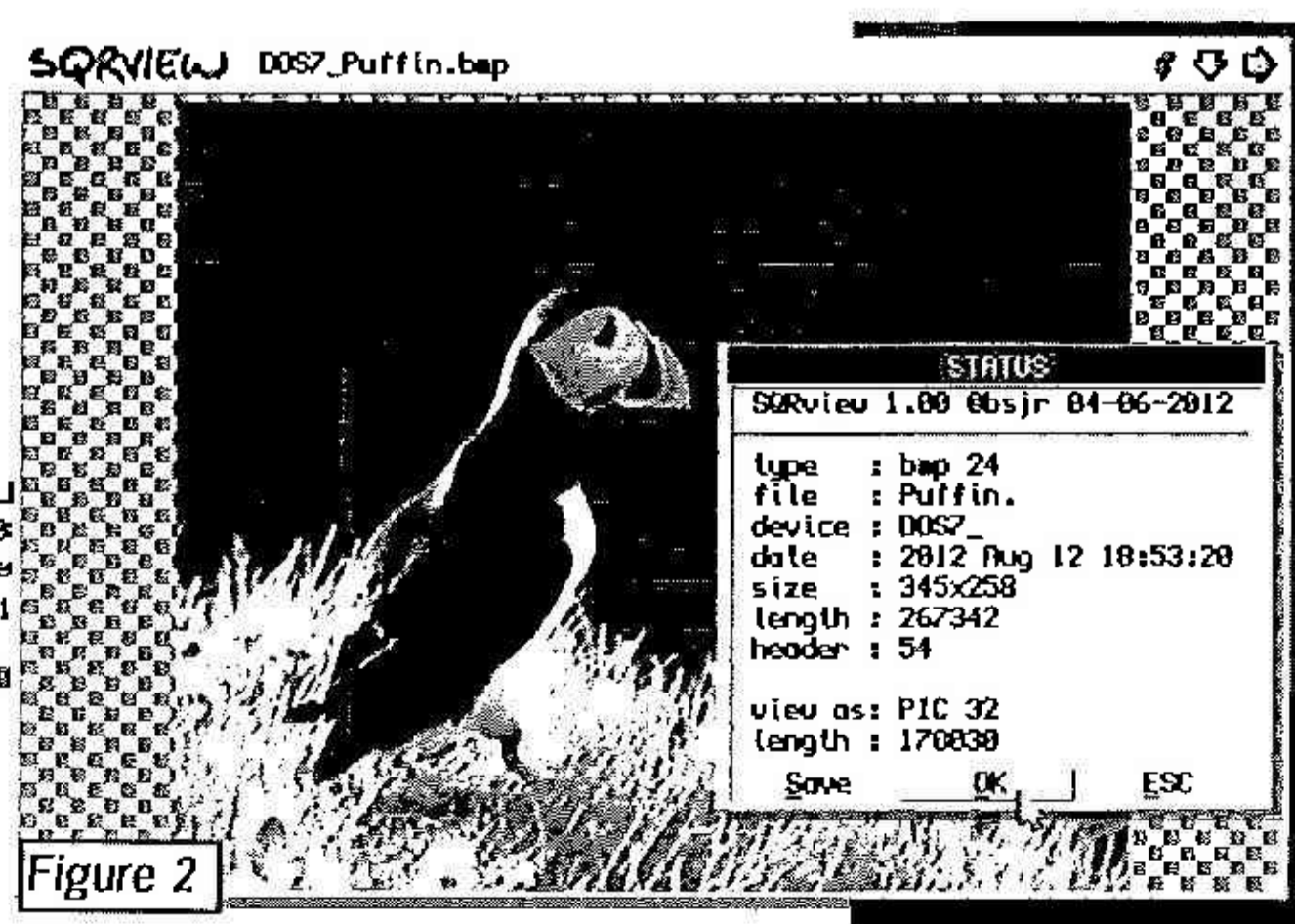


Figure 2



Figure 3

you only need to master those parts of the program needed for your QL system and way of working.

To run SQRview you have to have a system capable of displaying GD2 colours, but it can handle numerous different QL modes. You may also need a large amount of memory because in general graphics are memory hungry.

In summary it is impossible to describe the details of the program in a few hundred words. The best advice is to give it a try. You can download the program from Bob Spelten's website: <http://members.upc.nl/b.spelten/ql/>

numerous configuration options. Do not be put off by this apparent complexity as there is a comprehensive and easy to follow manual. Also

And let us hope that many more things will get out of hand, not only by Bob Spelten, but also many more developers.



Figure 4

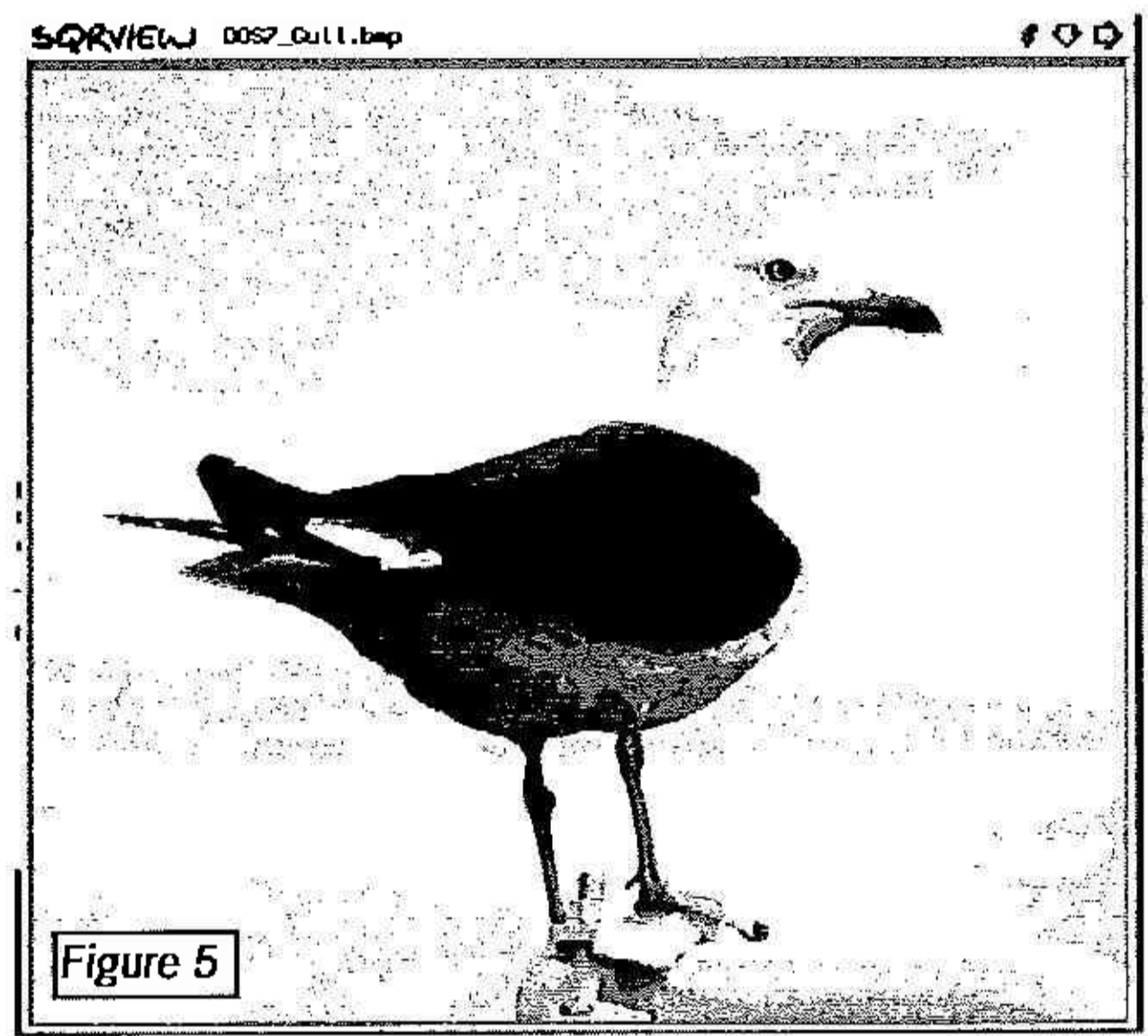


Figure 5

The continuing saga of George & Norman

by George Gwilt and Norman Dunbar

Norman's [ND] answers to George's [GG] comments on Assembler – Part 32. Comments inserted by the Editor are marked [ED].

[GG] Part 2 of Norman's articles on LibGen – Library Generator is rather tantalising in that after you have typed in all the code you get eight errors, as promised, when you try to assemble it. This of course is due to the missing routines required to input and process the files and comments on these will obviously have to wait.

[ND] Yes, I'm afraid that's very true. The article I submitted for publication did actually have the missing routines, unfortunately. Jochen contacted me to ask if he could split the article as he had run out of space. You may notice that the end of the article was on the inside back cover!

Anyway, I advised on a half decent place to split it which was right after the comment about getting assembly errors at this stage. There were a few more pages of code and editorial to follow, but these will be in the next issue, I hope.

[ED] Our apologies to Norman for having to split his article. It was only at the laying out stage that we discovered the article was longer than expected and by then we had run out of space.

[GG] Nevertheless I do have a few remarks to make on the program so far.

1. The code on page 45 which clears the status area and sets four loose items to unavailable could be shortened to:

```

    lea    wst0,a1          Unchanged
;    movea.l  a1,a0        Deleted
    moveq  #wst0_e-wst0-1,d1  Unchanged
st_clr   clr.b   (a1,d1.w)  Altered
         dbf    d1,st_clr   Unchanged

;st_loose lea    ws_litem+li_libfile(a1),a0  Deleted
         moveq  #3,d1          Unchanged

st_unav  move.b  #wsi_unav,ws_litem+li_libfile(a1,d1.w) Altered
         dbf    d1,st_unav   Unchanged
```

This code has two advantages over the original: it does not use A0 and there are fewer instructions. The fact that the original code, at least the first few instructions, were mine shows that any implied criticism should be directed to me not Norman.

[ND] Ah well, at least it's not all my fault this time. :-) However, this does indeed save an address register, which might be necessary in some applications.

[GG] 2. At the foot of page 49 the instruction finding whether ESC was typed is:

```
sh_esc   cmpi.w   #27,d1
```

I diffidently suggest that "cmpi.b" might be better. This is because in reading a string with a terminator such as ESC you have to process one byte at a time. Even if D1.B is #27 there may be no guarantee that D1.W is.

[ND] Hmm, I thought George had found a bug, but I checked my own code in iw_input and it sets D1.W to 27 if the ESC key is pressed to terminate the edit, so the CPM1.W is correct.

Equally, I checked the docs for the underlying WMENAME vector, and it too states that the returned terminating keypress value is in D1.W and not D1.B.

My own code sets D0 and D1.W correctly according to the docs which WMAN doesn't actually do. I know this as I had problems in this very utility because of pressing ESC and I traced the execution. My forthcoming code has the following in it:

```

    ...
    jsr  wm_ename(a2)      Edit string in info window.
    blt.s iw_exit         Negative is an error.

;-----
; Bug alert. It seems at present, that this vector always returns with
; D0 set to zero or negative, but never positive. Sigh. The following
; code tries to reset that situation to how it should be, according to
; the docs.
;-----

    cmpi.w #27,d1         ESC pressed = abort edit.
    beq.s iw_esc         Yes, all done.
    ...
```



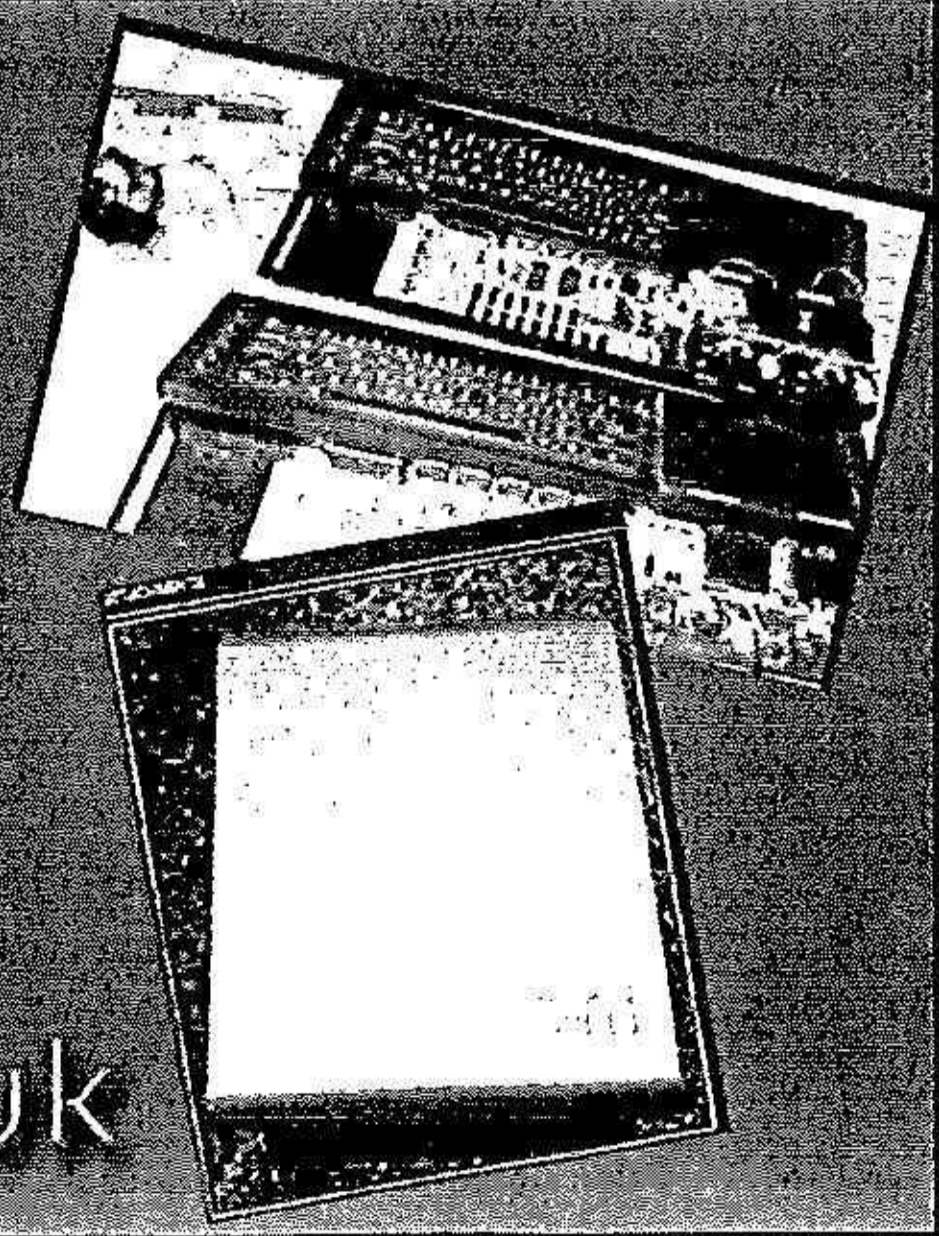
The Sinclair QL Forum
A Place for Sinclair QL Users to Meet!

QL FORUM

View all topics

GENERAL		TOPICS	POSTS	LAST POST
	The Welcome Area Introduce yourself here!	23	163	by igorstellar G Fri Feb 10, 2012 3:06 pm
	Help for New Users A section for new users to ask for help and also for people to post helpful tips.	14	87	by olliraa G Mon Feb 06, 2012 12:59 pm
	General QL Chat A place to discuss general QL issues.	50	336	by [user] D Fri Feb 10, 2012 3:31 pm
	Hardware Nagging hardware related question? Post here!	70	589	by thorsindair D Fri Feb 10, 2012 12:35 pm
	Software Anything software related	6	21	by [user] D Thu Feb 09, 2012 7:21 pm
	QL Emulation Discussion and advice about emulating the QL on other machines.	22	122	by Brane2 G Sun Feb 05, 2012 10:23 pm
	Compatibles Corry An area to discuss anything related to QL compatible computers such as the Thor, Aurora, 50 and 360	2	25	by [user] D Fri Jan 07, 2012 11:25 pm
MARKETPLACE		TOPICS	POSTS	LAST POST
	For Sale Sell your QL items here!	19	81	by RWAP G Thu Jan 19, 2012 5:26 pm

The online community
for all things ql



www.qlforum.co.uk

```

    cmpi.w #$0a,d1      Was ENTER pressed to end the edit?
    bne.s iw_esc       No, set D0 positive.

    clr.l d0           Zero = ENTER was pressed.
    bra.s iw_exit      Done.

iw_esc  moveq #1,d0     Set D0 positive as required.
    ...

iw_exit ...
    tst.l d0           Make sure Z is set correctly.
    rts

```

So, my library routine `iw_input` returns `D1W` and `D0` correctly according to the docs for `WM.ENAME/WN.RNAME` which is more than can be said for `WMAN`!

[GG] 3. The code at `sh_ok` on page 50 starts by saving the `WMAN` vector on the stack. There is actually no need for this since the vector has been stored at `wmvec(a6)` shortly after `start`. When the vector has to be restored you can use:

```
movea.l  wmvec(a6),a2
```

[ND] This is correct. I had forgotten that the `WMAN` vector was saved in the job's data area right back at the start.

[GG] I am always slightly nervous when I put something on the stack since I have to remember to clear the stack at every exit from the code. Using `wmvec(a6)` instead would make me that much more confident.

[ND] This is true, I do a similar thing myself from time to time. It's sometimes a lot easier on the old brain cells to stuff a much wanted value into the data area rather than trying to (a) remember to save it on the stack when the register is needed elsewhere and (b) restore it afterwards. Just use the register and get it from the data area when needed again.

I sometimes find that some of my code loads the desired value into the desired register when it is already there, but no harm done, other than wasting a few clock cycles.

[GG] 4. Finally I couldn't resist using my own versions of the missing routines and I can confirm that the file names are indeed printed as described by Norman.

[ND] I know! I've tested it! It's coming along nicely I think! It's quite disconcerting to see exactly how much extra hard work goes in to writing a utility in assembler compared to the code required to do the same in SuperBasic. I have a reasonably short utility – my proof of concept – written in SuperBasic and EasyMenu. Took me a couple of hours to cobble together and test. Assembly takes much much longer.

[ED] The discussion continued a few days later:

[GG] On exit from `WM.ENAME` and `WM.RNAME` the condition codes are set negative for an error (because `D0` is so set), zero for `NL` (ENTER) and positive non-zero for `UP/DOWN` and `ESC`. If there is no error `D0` can probably be anything. The non negative condition codes are set because of the value in `D1`. Your `WIKI` is correct in its description of the vectors. Interestingly it does not say what the value of `D0` is on exit!

[ND] Actually, I checked the docs. Every "error return" section is what we get back in `D0`. The error return for `wm.rname/w.ename` states that this will be +ve if the terminating character is not ENTER.

[GG] The manual is ambiguous. When it says "Any I/O sub system errors" it implies that if there is an error `D0` will contain its value (negative) and that the condition codes will be so set, as is normally the

case. It then appends the line '0 if terminating character not 'N' This only applies to the condition codes (but how can you tell).

[ND] The text mentions that the condition codes will be -ve (error), zero (Enter) or +ve (ESC, UP, DOWN), not D0. The table of registers gives (me) the impression it's the error return in D0 that is -ve, 0 or +ve.

[GG] I looked at the source code and found that D0 is set if there is an error, and the condition codes are set accordingly. If there is not an error the value in D0 is whatever it was sometime back in the program. I didn't check what this might be, but it could be anything. When there is not an error the terminating character is set in D1 and compared with \$0A thus setting the condition codes which, in that case, can't be negative.

The Games that Time forgot

by Peter Scott

I have an endless thirst for all things computer gaming, but the games that often fascinate me the most are the ones that get away, the games that time forgot. The obscure games where you read a little snippet or see a screen shot in an old QL Magazine. Just how many copies did it sell? Does someone have a copy quietly sleeping in a drawer or in a box up the loft? Or sadly have they all been binned and cease to exist? How did it play? What was the box like? Did it even get released in the first place?

I always like to imagine that even the rarest of games is out there somewhere waiting to be rediscovered. I recently got very lucky in tracking down a copy of Fleet Tactical Command but I'm still panicking about how to make a backup copy that keeps the games copy protection intact. Anyway, the following is a very shaky and certainly not complete feature on some games as mentioned in old magazines, fanzines or the rwap ql_Wiki that I'd love to know more about.

THE HEART OF GERN from PCBS was to be the first game in a series of graphical adventures set in the fantasy world of Princelands. The lone adventurer Tirac seeks to find a black jewel which imprisons an evil spirit. Said evil spirit is also sought by an evil cult as part of a master plan hence the plot is a race against time. The game was advertised (QL World May '87) on microdrive and disc but also for modes 4 or 8. Also mentioned is a sequel called THE FATE OF THE ORACLE, did the sequel see the light of day?

GUMSHOE LOGIC by Megacycal Software Ltd in 1985 is a strange sounding prospect of a game. Part text adventure and part

puzzle game which sees you running a Detective Agency. Money can be spent on informers to help solve the case. It conjures up images of a hard smoking/drinking Sam Spade but I have no idea if that is close to the mark or not.

AMBITION from WAY AHEAD was advertised (QL World August '88) as a business oriented strategy game. Possibly sounds like some kind of board or puzzle game maybe a computer version of monopoly. The QL had quite a few business games, Financial Climes and Tycoon are other examples of the genre.

THE SIMULATOR by Ekotek of Holland sounds like it was the most advanced of all the QL Flight Sims. One happy reader writing in to QL World '89 Sept had enjoyed flying under bridges, across lakes and round mountains. The game consisted of two worlds one in colour and one in monochrome plus an add-on pack was available with more worlds for £10

And finally two relatively unknown football games **TOP TEAM** by Arunsoft sounds like an accomplished football manager game featuring tactical selections, scouting reports and even real time text game commentary. The more unusual sounding **HERE WE GO** from Impact Entertainments was a text adventure written in using ACT (Adventure Creation Tool) that saw you tasked with getting to Wembley on cup final day.

Hopefully someone can remember these games or even better have a copy nestled away that would be something special. The RWAP QL Wiki has a list of games that are missing in action and

it's sad to think of these enticing games (and many others) that could be lost. Could the QL scene benefit in following the Spectrum scene which has well orchestrated projects to preserve software for the present and future?

[Ed.]: Peter asks for more information on these games, and, even as a non-games enthusiast, I can start the ball rolling.

THE SIMULATOR is about the only game that I have ever bought and I would agree with the reader who reviewed it in QL World. It was originally distributed on two microdrives or, later, on one floppy, and had copy protection allowing you to make no more than five copies. In addi-

tion to the default world the add on pack allowed you to fly by night or in urban, rural, sea and desert worlds. It could only run on native hardware because, if I remember rightly, of system pokes.

Eventually my native hardware became more ancient and erratic and I donated my copy to Rich Mellor for his preservation project. I believe Rich managed to get it running on one of the emulators but I can no longer remember the details.

Is there a reader who can remember more about THE SIMULATOR or one of the other programs Peter mentions?

What went wrong?

by Geoff Wicks

In the news section we report on Quanta's decision to scrap its plans for an active website, and to go over to a largely static one. It is the latest development in a long history of Quanta web failures. The tragedy is that for the last five years - and possibly even longer - Quanta was 90% on the way to achieving a successful active website. It has never mastered the last 10% - the art of keeping a website up to date. I hear more moans from Quanta members about out of date websites than on any other issue.

Quanta's first website was set up about 10 years ago by the then chairman, Robin Barker. Robin had considerable web skills and came the nearest to setting up a dedicated members' area. As a first stage he had created a page where members could register their contact details. Unfortunately there was less awareness in those days of the need for firewalls and the site became a target for hackers and spammers. When the host ceased trading it had to be taken down.

In the 8 years since Robin Barker stepped down as chairman Quanta has never had an officer with web experience or web expertise other than for one year when the present webmaster doubled up as treasurer. The new officers continued to parrot Robin's plans for the site promising a closed members' area containing among other things a members' forum. They had little understanding of the practical and technical issues involved and assumed a webmaster would have an instant solution to any problems. In particular they did not seem to understand the difference between a webmaster and a web editor.

As long ago as 2004, when I was on the Quanta committee, I asked who was responsible for the editorial content of the website and I never received a reply. I have asked the same question on many occasions since and usually the response has been a blank stare. In retrospect I think Quanta's officers genuinely did not understand the point of the question.

In the middle of the last decade I had personal experience of the officers' ignorance of the immediacy of the internet when I hosted dedicated pages for the QL2004 and QLis21 shows on my website. Usually I was able to update the non-Quanta QL2004 page within 24 hours, but updating the Quanta QLis21 page could take up to a week while I waited for the officers' approval.

The origins of the present website go back to 2007 when Quanta found someone to fill the post of webmaster that had been vacant for two years. He created a relatively simple site with Home, Events, Magazine, Subgroups, QL Systems and Traders pages. There was a major innovation at the end of 2008 when it became possible to pay the Quanta subscription online via the website and Paypal.

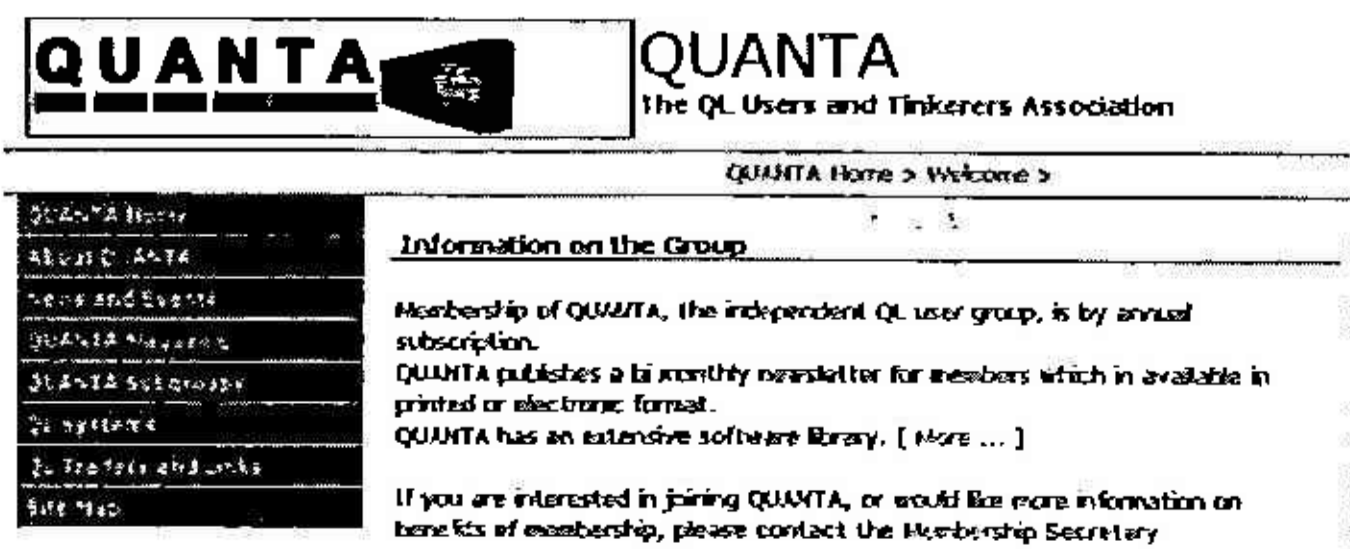


Fig. 1: Quanta's 2007 website

The website soon fell into disrepute for out of date content. The officers had no machinery for regular updating and what updating there was appeared to be on the occasional whim of the chairman. (I shall draw a discrete veil over one updating in 2008 that had to be taken down very quickly for legal reasons.)

During 2009 the website was upgraded with an open members' area, mainly to allow downloading of programs published in the Quanta Magazine, but also containing a QL news section. Quanta again announced the intention to introduce a closed area where members could download programs from the Quanta library and the latest copies of the Quanta Magazine. A members' forum would also be introduced. By then an extra member had been co-opted onto the committee to assist in its development.

Quanta's two webmasters soon concluded that further development of the site would not be possible unless the work was shared by other committee members. They recommended the use of a Content Management System (CMS). Quanta looked at two systems, Joomla and Typo3, and at a meeting held in late 2009 the committee opted for the latter.

The implementation of the CMS proved to be more difficult than expected and for over six months during 2010 it proved technically impossible to update the site. As often happens when Quanta is faced with a serious problem, the officers went into Trappist mode and withheld the problems from the members. The only information given was a cryptic message from the secretary in the Dec 2010/Jan 2011 Quanta Magazine:

'Taking about the website, I think we are close to being ready to launch on this, more information will be found in this issue.'

In fact there was no further information in that issue.

By the last quarter of 2010 QL Today had been aware for some time that the Quanta website was in serious technical trouble. We initially deduced it from the continued failure to update the news section and later had some confirmation from Quanta sources. We decided, however, not to publish, partly because no Quanta committee member would speak on the record and partly because we knew Quanta had an emergency committee meeting planned to attempt to solve the problems.

Two months after the committee meeting there was still no sign of the problems being solved,

and, after some critical comments about the website on the QL-users email list, we published the story in the Feb-Apr 2011 issue. What I did not publish, however, was that the web problems had caused serious tensions within the Quanta committee with some angry emails in circulation. I formed the impression that more than one member of the committee were considering their position.

News

QUANTA'S Web Collapse

The Quanta website has been hit by serious technical problems that have prevented it from being updated for over 6 months. QL Today understands the new Content Management System (CMS), introduced at the beginning of last year has suffered a massive failure. The cause of this is proving elusive. Individual Quanta committee members are reluctant to speak on the record about the problems, but a number of suggestions for the failure are being made. These include Quanta's aspirations for the site being too ambitious; the lack of technical support from a free host; and the difficulties of integrating a Linux based CMS into a Windows environment.

Alison Souther
the Quanta Ma
Talking about
close to being
information will
There was no
the magazine
signs of the pr
Quanta has ha
has never succ
the past the p
who had respo
the site and th
come this prob
well advanced,
creating it here

Fig. 2: QL Today's 2011 news story

At least one member argued for Quanta to go over to a simpler website, but the advice of the webmasters prevailed. The problem was that a free web host could not offer the technical support that Quanta needed. A new website was launched using a paid host in the middle of 2011, but the new web host was very expensive. In 2011 Quanta paid £346 for its website, more than seven times what I pay for the Just Words! website.

The webmasters justified this expenditure in the March 2011 Quanta Magazine:

'The WEC service includes daily backups of the QUANTA website and members' area, automatic upgrades of Typo 3 software modules, access to online training materials, and a user forum so we can share best practice with other Typo 3 users around the world.

WEC have developed an impressive templating engine, which will enable us to alter the look and feel of the QUANTA website by changing a few background images and style sheets.

I am developing different templates for the QUANTA homepage, each of the landing pages, and the detailed content pages.

Best of all, WEC have written some custom plug-in modules:

- A threaded discussion forum for the members' area,
- A Google map plugin so we can show the

location of QUANTA subgroups, workshop venues and events

- A volunteering module to enable members to indicate their areas of interest and ability to help
- A mailing list module to enable us to circulate a regular e-newsletter to members
- Interactive photo galleries, so we can promote QUANTA events and workshops

Additional features that will be provided:

- Members Area, archived copies of magazines and source code will be available for members to download
- News updated frequently, news can be submitted to the committee for publication on the website or when the committee spots some important news it will be published
- Multiple editors, additional editors can be brought online to edit content at any time.

A year later the webmasters had a different story to tell:

"The web site has been pretty static for the last 12 months with very little activity."

and

"...hopefully you will see some changes start to happen soon and the members' area will become accessible."

8 months after this promise was made no changes were apparent. It was no longer a technical failure, but a human one. In their Quanta Magazine article the webmasters wrote the new web host would "free up the committees' (sic) time to focus on content and design, rather than worrying about the complexity of this powerful CMS", but in practice only one member of the committee other than the webmasters had been prepared to use the CMS. Not one of the officers had contributed to the website's content.

At the beginning of November the website was describing a magazine published 18 months previously as 'recent'. The shows page was still publicising the Austrian show held in June and although Quanta has had a new constitution for 7 months, the old one remained on the site.

In the middle of October the Quanta committee held a further meeting to discuss the website. In a message posted on the QL-users email group the webmaster reported:

"The committee decided that moving to a cheaper platform was a good idea and I have now been tasked with finding a service to use (I

know there are plenty around so will be feeding back my thoughts to the committee). It was also agreed that the new site will be mainly static so the committee will not be making much effort in updating the site (though we will use a different CMS system we can make updates as required)."

Just what this means in practice is unclear, but it appears that Quanta has abandoned plans for a closed members' area and that site will become purely information on how to join Quanta and on QL systems.

In my opinion the closed members' area was always a non-starter. It was proposed that members could download programs from the Quanta library, but these days traffic in and out of the library is negligible. The members could also download the latest Quanta Magazines, but why would they need to do this when they get a copy anyway? It is also current policy to make the magazine public after 2 years.

The only purpose of a closed members' area would be a members' forum, but Quanta had made no detailed plans for this. The issue of moderation of a forum had not even been discussed and the time required for moderation had not been considered.

In fact once you scrap the idea of a closed members' area, the present website is a good one provided it is kept up to date. That would not be an onerous task. The hardest job is keeping the news section up to date, but that is already working satisfactorily. Next would be the magazine with six updates a year and then the events page with no more than 2 or 3 shows a year.

Ultimately the failure of Quanta websites has not been technical, but human. And most of the blame must go not on the shoulders of the webmasters but on the officers. The content of the website is not the responsibility of the webmaster, but that of the officers and they have always failed in their duty over many years to keep websites up to date.

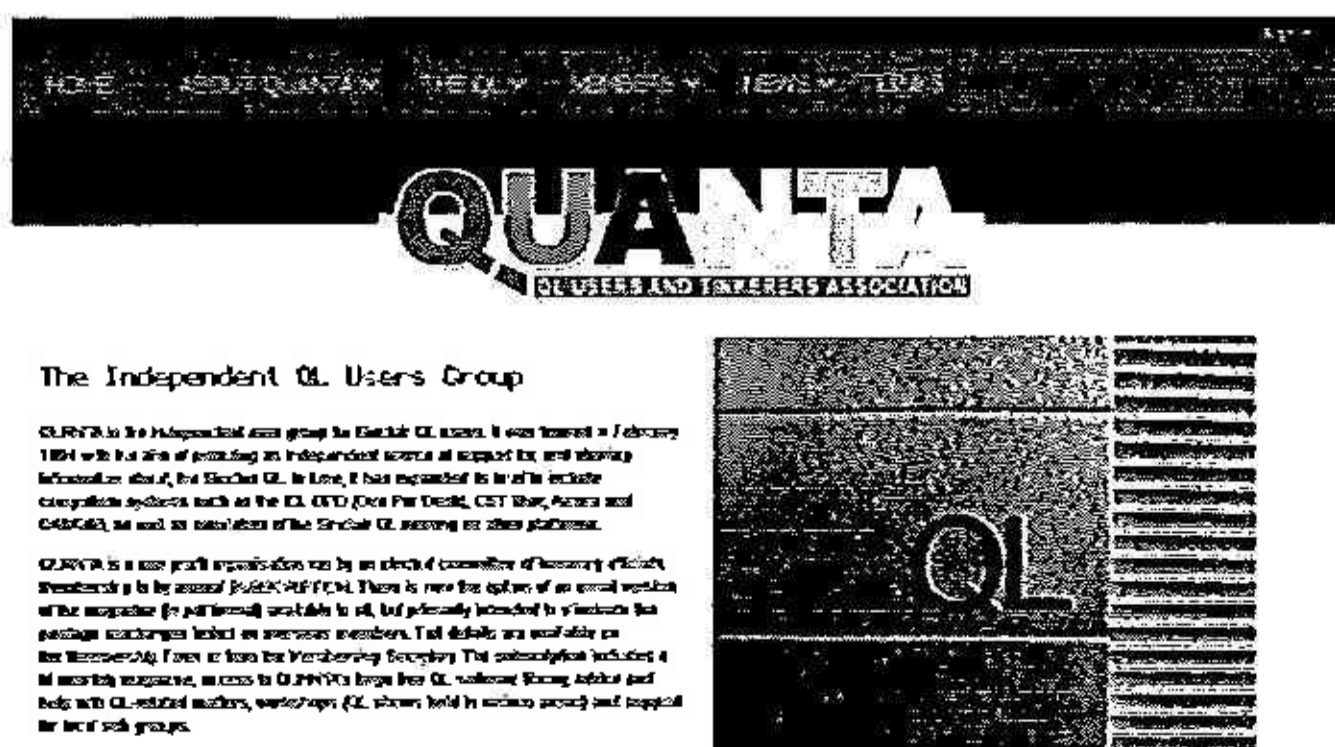


Fig. 3: Quanta's present website

QUO VADIS
DESIGN

Independent Information
Technology Services

www.ql-qvd.com

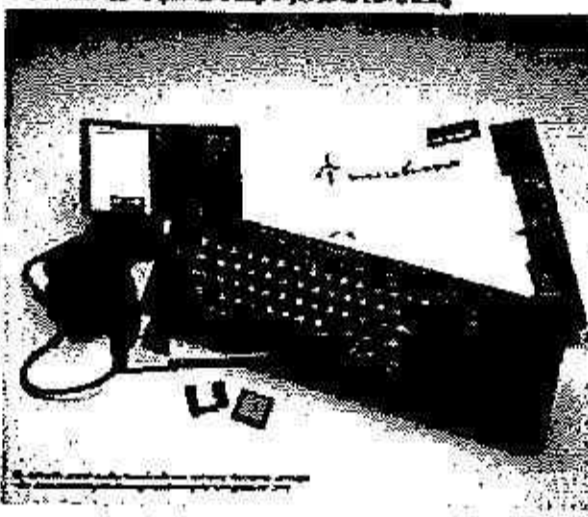
QL/QDOS/SMSQ/E Software

QUO VADIS
DESIGN Independent Information
Technology Services

QL/QDOS/SMSQ/E Software

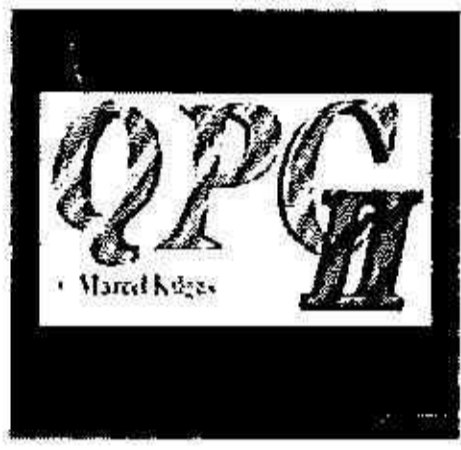
Home | Products | Support | Company | Contact

Welcome
Quo Vadis Design sells software for the Sinclair Quantum Leap computer (QL) and variants including a new OS called SMSQ/E.
The QL is a computer in its 25th year Anniversary.
The Sinclair QL - a quantum leap in personal computing



News
QVD QL News Blog - keep up to date
News Blog
24/02/2009
Quo Vadis Design Website Launched
01/02/2009

FEATURED PRODUCT



BUY NOW!

Software emulations of the QL now exist which can run on a PC/Mac with Windows/Linux or Mac Operating systems.

Copyright © 2009 Quo Vadis Design. All Rights Reserved. Home | Products | Support | Company | Contact | Privacy

ALL THE BEST FOR THE NEW YEAR

Bruce@ql-qvd.com

Quo Vadis Design
38 Derham Gardens
Upminster
RM14 3HA
UK

Tel: +44 (0)20 71930539
Fax: +44 (0)870 0568755

Check the QL News Blog on
our website for updates.
www.ql-qvd.com/blog

QL Today The QL magazine
for all QDOS, QL,
SMSQ ... users!

Subscriptions taken online

Machin's Formula for PI

by George Gwilt

In my recent article on large integers I quoted Machin's formula for PI as

$$4*PI = 4*atan(1/5) - atan(1/239).$$

$4*PI$ is wrong. It should have been $PI/4$.

Happily the code showing the calculation is actually correct. But how on earth is such a simple expression equal to $PI/4$? I give the following explanation in case others are as puzzled as I was.

The problem is to find a more swiftly converging series than that for

$$PI/4 = atan(1).$$

Suppose we find a small angle A whose tangent is the reciprocal of an integer n and is nearly equal to $PI/(4*k)$ where k is an integer. In that case we would have

$$A = atan(1/n) \quad [1]$$

and

$$PI/4 = k*A - B \text{ where } B \text{ is a small angle. Then}$$

$$B = k*A - PI/4 \quad [2]$$

So

$$\tan B = \tan(kA - PI/4) \quad [3]$$

Now

$$\tan(x+y) = (\tan x + \tan y) / (1 - \tan x * \tan y) \quad [4]$$

From this we see that

$$\tan 2x = 2 \tan x / (1 - \tan x^2) \quad [5]$$

Using 4 on 3 gives

$$\tan B = (\tan kA - 1) / (\tan kA + 1) \quad [6]$$

Repeated use of [4] and [5] will give $\tan B$ in terms of $\tan A$. If $\tan A$ is a rational number so will $\tan B$ be. Thus both A and B can be found by series more rapidly converging than that for $atan(1)$.

Examples

1. $n = 2$ and $k = 2$

$$\tan A = 1/2 \text{ so } \tan 2A = 1/(1-1/4) = 4/3. \text{ Thus } \tan B = (4/3 - 1)/(4/3 + 1) = 1/7.$$

So

$$PI/4 = 2atan(1/2) - atan(1/7). \quad [7]$$

2. $n = 3$ and $k = 2$

$$\tan A = 1/3 \text{ so } \tan 2A = (2/3)/(1 - 1/9) = 3/4.$$

$$\text{Thus } \tan B = (3/4 - 1)/(3/4 + 1) = -1/7. \text{ [B is negative]}$$

So

$$PI/4 = 2atan(1/3) + atan(1/7). \quad [8]$$

3. $n = 5$ and $k = 4$

$$\tan A = 1/5 \text{ so } \tan 2A = (2/5)/(1-1/25) = 5/12. \text{ Applying [5] again we get}$$

$$\tan 4A = (10/12)/(1 - 25/144) = 120/119.$$

Thus $\tan B = (120/119 - 1)/(120/119 + 1) = 1/239$.

So

$$\text{PI}/4 = 4\text{atan}(1/5) - \text{atan}(1/239). \quad [9]$$

4.

Taking the average of examples 1 and 2 gives

$$\text{PI}/4 = \text{atan}(1/2) + \text{atan}(1/3) \quad [10]$$

[9] is Machin's formula.

Programming in Assembler, Part 31

LibGen - Library Generator - Part 3

by Norman Dunbar

Ed: Thanks a lot for your patience - as explained in the previous issue, we ran out of space, therefore we had to split this article into two parts.

The Utilities Library

Scroll to the end of the file and add the following lines, obviously, you will adjust the file name to suit your own installation. Unless, of course, your setup is exactly the same as mine. You may also note, that while I refer to the code as a library, it is not yet converted into one. When we complete LibGen, we shall use it to build a working library. (Does that count as recursion?)

```
;  
; And finally finally, my own utilities.  
;
```

```
in win1_source_qltoday_pe_utilities_asm
```

The code above goes just after the part where we pull in George's sprite code. It is not possible for a reusable library's routines to make any assumptions about the use of registers outside of the library itself.

To this end, each of the following routines will preserve any register that it uses, with the exception of registers that are documented to return a potentially useful value. Create a new file with the name given above, in my case - yours may be different, and add the following routines to it.

The first subroutine is IW_INPUT which allows an information window to be used to accept input from the user. It works by calling the WMAN routine that sets the channel ID to the information window in question, then copies the current value of the string to be edited to a working buffer. A jump back into WMAN is made to allow the user to edit the string. The edit may be terminated by ENTER, ESC, Up arrow or Down arrow.

Note:

The QPTR documentation states that D0 will be zero if ENTER was pressed, negative for any errors, and positive if ESC, Up or Down were used to terminate the edit. This is not true as there is no code in WMAN that does this. D0 will always be zero at the end of an edit, unless any errors occurred. In order to be consistent with the documentation, the library routine iw_input correctly sets D0 to a positive value - actually 1 - when ESC, Up or Down arrow terminates the edit.

```

;=====
; This file contains useful utilities for a Pointer Environment
; application. It's just crying out to become a library! ;-)
;
; IW_INPUT  -- Get input from a designated information window.
; IW_PRINT  -- Print a string to a designated information window.
; CP_STRING -- Copy a string between two locations.
; AP_STRING -- Append one string to the end of another.
;=====

;=====
; IW_INPUT : accept input from an information window. The routine has
;           a working buffer of 1024 characters maximum which is
;           more than enough for any information window width.
;=====
; Entry Registers:
;
; D1.W Information window number.
; D2.L Ink colour, or negative ink colour. See WM_SWINF documentation.
; A2.L WMAN vector.
; A3.L Pointer to string.
; A4.L Pointer to work def.
;-----
; Exit Registers:
;
; D1.W Terminating character: Enter, Esc, Up arrow, Down arrow.
; D2.L Preserved.
; A1.L Buffer pointer.
; A2.L Preserved.
; A3.L Buffer pointer.
; A4.L Preserved.
;-----
; Errors:
;
; D0 = negative: Any I/O error. Old string at (A3) unaffected.
; D0 = zero:     Enter terminated the edit. Old string at (A3) updated.
; D0 = Positive: Up, Down or ESC terminated the edit. Old string at (A3)
;                updated, unless ESC pressed.
;=====
iw_input movem.l a2-a3, -(a7)      Save WMAN vector & source buffer.

iw_copy  lea iw_buffer, a2        Copy destination buffer.
        bsr cp_string           Copy to work buffer.
        move.l (a7), a2         Get WMAN vector.

        jsr wm_swinf(a2)        Set channel to info window.
        bne.s iw_exit          Bale out on error.

        lea iw_buffer, a1       Edit buffer required in A1.
        jsr wm_ename(a2)       Edit string in info window.
        blt.s iw_exit          Negative is an error.

;-----
; Bug alert. It seems at present, that this vector always returns with
; D0 set to zero or negative, but never positive. Sigh. The following
; code tries to reset that situation to how it should be, according to
; the docs.
;-----

        cmpi.w #27, d1          ESC pressed = abort edit.
        beq.s iw_esc           Yes, all done.

```

JOCHEM**MERZ****SOFTWARE**

**Kaiser-Wilhelm-Str. 302
47169 Duisburg
Germany**

**Phone +49 203 502013
Fax +49 203 502012
EMail: SMSQ@J-M-S.com**

XMAS OFFER - UNTIL FEB 2012! 10% DISCOUNT ON ALL PRODUCTS LISTED HERE!

QPC2 Version 3 + SMSQ/E Software QL-Emulator for PC's		EUR 59,90
QPC2 Version 3 - Upgrade from QPC2 Version 2		EUR 19,90
QPC2 Version 3 - Upgrade from QPC2 Version 1		EUR 39,90
QPC Print - printer emulation driver for QPC		EUR 39,90
Agenda Agenda program for WMAN and Prowess	[V1.09]	EUR 14,90
Suqcess Database front-end for WMAN	[V2.05]	EUR 19,90
QD2003 Pointer-Environment-Editor	[VB.01]	EUR 29,90
QD2003 Upgrade from Version 9 and older	[VB.01]	EUR 14,90
QMAKE Pointer-driven MAKE for GST/Quanta Assembler	[V4.31]	EUR 14,90
BASIC Linker	[V1.21]	EUR 14,90
WINED Floppy/Harddisk Sector- & File-Editor	[V1.26]	EUR 14,90
FiFi II File-Finder - Extremely useful!	[V4.31]	EUR 14,90
EPROM Manager	[V3.02]	EUR 14,90
QSpread2003 Spreadsheet Program	[V4.04]	EUR 29,90
QSpread2003 Upgrade from Version 3 and older	[V4.04]	EUR 14,90
QPAC I Utility programs	[V1.11]	EUR 19,90
QPAC II Files, Jobs & other Things	[V1.45]	EUR 29,90
QTYP II Spell checker	[V2.17]	EUR 19,90
QPTR Pointer Toolkit	[V0.30]	EUR 29,90
DISA Interactive Disassembler	[V3.04]	EUR 29,90
CueShell	[V2.14]	EUR 29,90
EasyPTR Version 4	[V4]	EUR 59,90
EasyPTR Version 4 - Upgrade from earlier versions	[V4]	EUR 39,90
QDT - QL Desktop program		EUR 59,90
QMENU Version 8 - with new, printed Manual	[V8.02]	EUR 24,90
QMENU Version 8 - Update from earlier Versions, also with printed manual		EUR 17,90

Please add EUR 4,90 for postage to all destinations - Germany, Europe, Worldwide!

**If you place your order via website SMSQ_J-M-S.com, the full price will be shown
but the 10% will be taken off manually by me after you placed your order!**

We accept VISA, MasterCard, JCB, Diners Club and American Express online and offline!

Details for money transfers:

- Deutschland: Jochen Merz, Account 493 50 431, Postbank Essen, BLZ 360 100 43
- Österreich: Jochen Merz, Account 85055317, PSK Wien, BLZ 60000
- Switzerland: Jochen Merz, Account 60-690080-4, PostFinance, Clearing-Nr. 09000
- The Netherlands: Jochen Merz, Gironummer 3258439, Postbank NL Amsterdam
- and from all other countries in EUR with IBAN and BIC to account
Jochen Merz, Deutsche Postbank AG, IBAN: DE21 3601 0043 0611 1004 37 / BIC: PBNKDEFF 360
- UK customers can pay in £ (convert EUR prices above to £ by multiplying with 0.83) to
Jochen Merz, Account 83795395, Citibank UK, Sort code 30-00-45
or send cheques in £ - no fee for UK sterling cheques!
- If you wish to pay via paypal, send money to Paypal@J-M-S.com

Cheques payable to Jochen Merz only!
Price list valid until 28th of Feb. 2013

```

        move.l a3,a2          Original buffer is destination now.
        lea iw_buffer,a3     New string found here.
        bsr cp_string        Copy new value to old buffer.

        cmpi.w #$0a,d1       Was ENTER pressed to end the edit?
        bne.s iw_esc         No, set D0 positive.
        clr.l d0             Zero = ENTER was pressed.
        bra.s iw_exit        Done.

iw_esc  moveq #1,d0          Set D0 positive as required.

iw_exit movem.l (a7)+,a2-a3  Tidy WMAN vector off stack.
        tst.l d0             Make sure Z flag is correct.
        rts

```

```

;-----
; Working buffer for IW_INPUT. A maximum of 1024 characters is allowed.
;-----
iw_buffer ds.w 512+1

```

The next routine is iw_print and simply prints a string to an information window. The information window is cleared first before printing. All the hard work is done by WMAN.

```

;=====
; IW_PRINT : Print a string to a designated information window.
;=====
; Entry Registers:
;
; D1.W Information window number.
; D2.L Ink colour or negative ink colour. See WM.SWINF documentation.
; A2.L WMAN vector.
; A3.L Pointer to string to be printed.
; A4.L Pointer to window working definition.
;-----
; Exit Registers:
;
; All registers are preserved except D0.
;-----
; Errors:
;
; Any I/O error.
;=====
iw_print movem.l d1-d3/a0-a1/a3,-(a7) Save working registers.
        jsr wm_swinf(a2)      Set channel to info window.
        bne.s iwp_exit        Bale out on error.

iwp_cls  moveq #$20,d0        CLS.
        moveq #-1,d3         Timeout.
        trap #3              Do it.

iwp_prnt moveq #io_sstrg,d0   Send a string of bytes.
        move.w (a3)+,d2       Byte count.
        beq.s iwp_exit        Nothing to do.
        moveq #-1,d3         Timeout.
        exg a3,a1             Pointer in A1 is needed.
        trap #3

iwp_exit movem.l (a7)+,d1-d3/a0-a1/a3 Restore working registers.
        tst.l d0
        rts

```


The subroutine `cp_string` copies a string from a location pointed to by `A3` to the location pointed to by `A2`. The two locations must be holding a QDOS formatted string with the word count at the beginning and the bytes immediately following.

The code below copies byte by byte so it best suited to fairly small strings and no checks are carried out to see if the strings overlap in memory, which could lead to corruption after the copy has completed. However, it is unlikely for this to actually happen when copying strings around.

```

;=====
; CP_STRING : Copy a string from a buffer at A3.L to a buffer at A2.L.
;           The word length is always copied, even if zero.
;           NOTE: No checks are done to prevent overlapping!
;=====
; Entry Registers:
;
; A2.L Destination buffer address.
; A3.L Source buffer address.
;-----
; Exit Registers:
;
; All registers, except D0, are preserved.
;-----
; Errors:
;
; None. D0 is zero on exit.
;=====
cp_string
    movem.l a2-a3,-(a7)      Save the workers.
    move.w (a3)+,d0         Get the source length.
    move.w d0,(a2)+         Copy to output buffer.
    beq.s cs_exit           Nothing more to do.

    bra.s cs_next

cs_loop  move.b (a3)+,(a2)+  Copy one byte.
cs_next dbf d0,cs_loop      And all the rest.

cs_exit  movem.l (a7)+,a2-a3 Restore the workers.
        clr.l d0           No errors.
        rts

```

And finally, for now at least, the `ap_string` routine appends the string at `A3` to the end of the string at `A2`.

It is the responsibility of the programmer to ensure that enough space exists at the end of the destination string to hold the string being appended. The routine cannot check for this and assumes all will be well.

If your buffer is too small, there is a good chance that whatever follows the buffer will be corrupted. You have been warned!

```

;=====
; AP_STRING : Append one string to another. The destination buffer is
;           assumed to be big enough for both strings. This code
;           cannot check for this.
;=====
; Entry Registers:
;
; A2.L Destination string address.
; A3.L String to be added to A2.
;-----

```

```

; Exit Registers:
;
; All registers, except D0, are preserved.
;-----
; Errors:
;
; None. D0 is zero on exit
;-----
ap_string
    movem.l d1/a2-a3,-(a7)    Save working registers.
    move.w (a3)+,d0          Get second string size.
    beq.s as_exit            Nothing to do, bale out.

    move.w (a2),d1           Current length of destination.
    add.w d0,(a2)+           Store the new length.
    adda.w d1,a2             End of destination string.
    bra.s as_next

as_loop  move.b (a3)+,(a2)+   Append one byte.
as_next dbf d0,as_loop       And all the rest.

as_exit  movem.l (a7)+,d1/a2-a3 Restore working registers.
        clr.l d0             No errors.
        rts

```

For now, those are all you need.

Now when you assemble to main file, it will hopefully assemble without errors and you will be able to run the program.

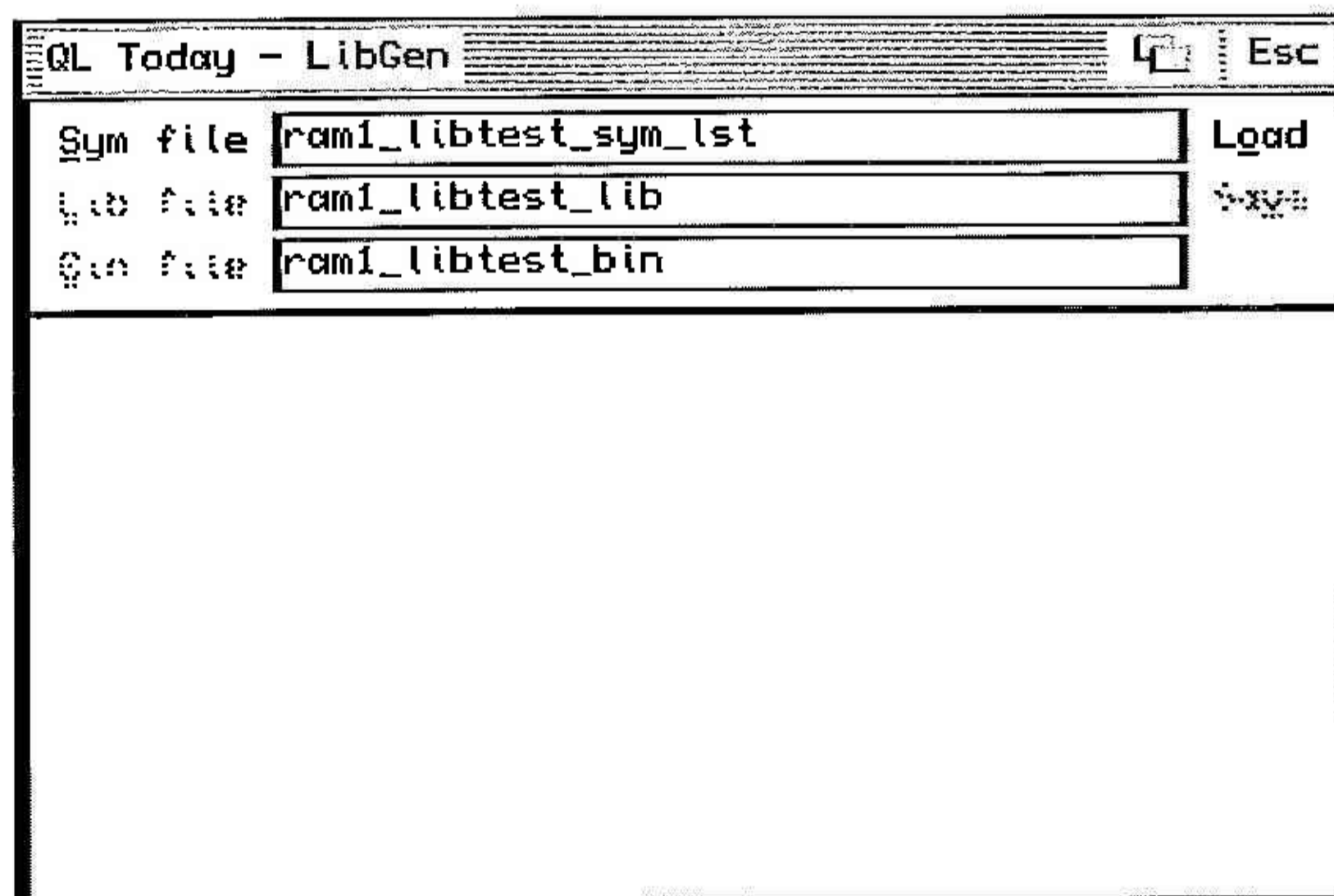
Once you type in a filename which is longer than 7 characters, you will hopefully see it transferred to the library and binary filenames but each with a separate extension.

Under normal circumstances, these defaults will be correct, but you will be given the opportunity to change them when the symbol file has been loaded.

In order to load the symbol file, you hit the "Load" loose item, or press 'O', but at the moment, this loose item is not yet wired in, so does nothing. If you have a slow QL, you might see the status change from available to selected and back to available again.

Pressing the TAB key forces the pointer to jump into the application window, and pressing ESC exits the program.

Hopefully it will look remarkably similar to the following screen shot.



We have now almost completed step 3. It is now time to complete step 3 and move on to step 4. Change the code for the "Sym file" loose item's hit routine to the following.

```
;  
; SYM FILE loose item action routine.  
;  
afun0_2  movem.l d5-d7/a0-a4,-(a7) Preserve important registers.  
        bsr.s li_unav          Make LI status unavailable.  
        bsr sym_hit           Do it all.  
        movem.l (a7)+,d5-d7/a0-a4 Restore important registers.  
        cmpi.w #27,d1         Did we abort the edit?  
        beq li_reset         Yes, bale out.  
        bsr.s li_avail        Make LI status available, if all ok.  
        bra li_reset         Make Sym file available.
```

The changes are quite simple. We call out to a subroutine - li_unav - to change the status for all loose items from "Lib file" onwards to become unavailable. As we are potentially changing the symbol file name in this action routine, we cannot allow incorrect data to be used by the Library and Binary file names.

After returning from the main sym_hit code, we check to see if the user aborted the edit with the ESC key. If so, then we know that the symbol file name currently on display is potentially garbage, and we exit leaving only the "Sym file" loose item available.

If the user successfully entered a symbol file name and did not abort the edit, we call the routine at li_avail to set all the loose items, except "Save" to available. "Save" will be enabled by the actions of the "Load" loose item later.

```
;  
; Reset all Loose Items, affected by hitting "Sym file" to unavailable  
; which is only possible, using the following code, because the 4  
; bytes affected are consecutive.  
;  
li_unav  move.l #$11111111,ws_litem+li_libfile(a1)  
        bra.s li_rdrw          Redraw selected loose items.  
  
li_avail move.l #$01011101,ws_litem+li_libfile(a1)  
li_rdrw  moveq #-1,d3          Request selective redraw.  
        jsr wm_ldraw(a2)      Redraw selected loose items.  
        rts
```

The above code is all that is required to enable and disable the affected loose items. It is quite simple to do this as there are only 4 loose items affected and the status bytes for those four are consecutive in the status area. Also, for efficiency, we set all 4 as desired, and make one single call to wm_ldraw rather than one call per loose item.

Note:

You may remember that I mentioned, way back at the start, that there was more than one way to set the status bytes for these 4 loose items? Well, the code above is the other way.

You may now assemble and run the application, HIT "Sym file" and type something in.

If you abort the edit with ESC then the loose items remain disabled. If you press ENTER (or up/down arrows) then the loose items are enabled. "Save" remains disabled at all times.

Handling the Lib File Loose Item

There's not much happens when the user HITs the "Lib file" loose item. The user is allowed to change the default name, chosen for the library file, from that defaulted by LibGen. The defaults for this option and the binary file are based on the original symbol file which itself is based on the original source file name.

For example, if you assemble a source file named ram1_libtest_asm, you will get a generated binary file of ram1_libtest_bin, a listing of the code in ram1_libtest.lst and a symbol file of ram1_libtest_sym.

Using George's sym_bin utility, you convert ram1_libtest_sym into ram1_libtest_sym.lst. It is this latter file that LibGen processes for you.

Based on the name of this latter file, LibGen chooses ram1_libtest_lib for the library file – the one containing an 'IN' and a 'BIN' instruction, and for the actual binary file making up the library, chooses ram1_libtest_bin.

In most cases, these defaults will be fine, but at least LibGen gives you the ability to change them as desired.

After all, it's not wise to keep your various library files on ram1_ is it!

The code for the 'Lib file' hit routine is as follows. This replaces the existing dummy code already in the main source file.

```
;  
; LIB FILE loose item action routine.  
;  
afun0_3  movem.l d5-d7/a0-a4, -(a7) Preserve important registers.  
        bsr lib_hit                Do it all.  
        movem.l (a7)+, d5-d7/a0-a4 Restore important registers.  
        bra li_reset              Make Lib file available.
```

As before, I tend to prefer the actual hit routine to be small. The above code simply preserves the desired registers, calls out to the lib_hit code, then restores the registers and exits resetting the 'Lib file' loose item to available. You will see that it remains in a selected state until you complete the edit.

```
;  
; This code carries out all the nasty work for a hit on the Lib file  
; loose item. It is called from afun0_3 above.  
;  
lib_hit  moveq #iw_libfile, d1      Info window number in d1.w.  
        lea lib_buffer, a3         Current lib file buffer.  
        moveq #1, d2              Blue Ink when editing.  
        bsr iw_input              Get input from desired info window.  
        blt.s lf_exit             Something went wrong, bale out.
```

Nothing much of interest here, the processing is almost identical to that we have seen already for allowing the user the ability to type a symbol file name.

We don't have to check for the ESC key terminating the edit because we always, unless there was an error, exit via the tidy up code at lf_ok.

Remember that our library routine iw_input copies the data from the working edit buffer back to the correct location on a successful edit.

```
;  
; Print the lib file name. We do this at the end of a normal edit and  
; when the user aborts with ESC. This keeps the info window tidy.  
;  
lf_ok    moveq #iw_libfile, d1      Information window desired.  
        moveq #0, d2              Black ink.  
        lea lib_buffer, a3         Filename to print.  
        bsr iw_print              Print it.
```

lf_exit rts

The code above, you may notice, doesn't preserve the terminating character in D1. There isn't really any requirement to do so in the handling of a 'Lib file' HIT.

Handling the Bin File Loose Item

The code for handling a hit on the 'Bin file' loose item is almost identical to the above. It will be shown here without further discussion.

```
;-----  
; BIN FILE loose item action routine.  
;-----  
afun0_6  movem.l d5-d7/a0-a4,-(a7) Preserve important registers.  
        bsr bin_hit                Do it all.  
        movem.l (a7)+,d5-d7/a0-a4 Restore important registers.  
        bra li_reset              Make Bin file available.  
  
;-----  
; This code carries out all the nasty work for a hit on the Bin file  
; loose item. It is called from afun0_6 above.  
;-----  
bin_hit  moveq #iw_binfile,d1      Info window number in d1.w.  
        lea bin_buffer,a3         Current bin file buffer.  
        moveq #1,d2              Blue ink when editing.  
        bsr iw_input             Get input from desired info window.  
        blt.s bf_exit           Something went wrong, bale out.  
  
;-----  
; Print the bin file name. We do this at the end of a normal edit and  
; when the user aborts with ESC. This keeps the info window tidy.  
;-----  
bf_ok    moveq #iw_binfile,d1      Information window desired.  
        moveq #0,d2              Black ink.  
        lea bin_buffer,a3         Filename to print.  
        bsr iw_print             Print it.  
  
bf_exit rts
```

If you assemble the code and execute it you should find that everything works as desired.

End Of Chapter 32

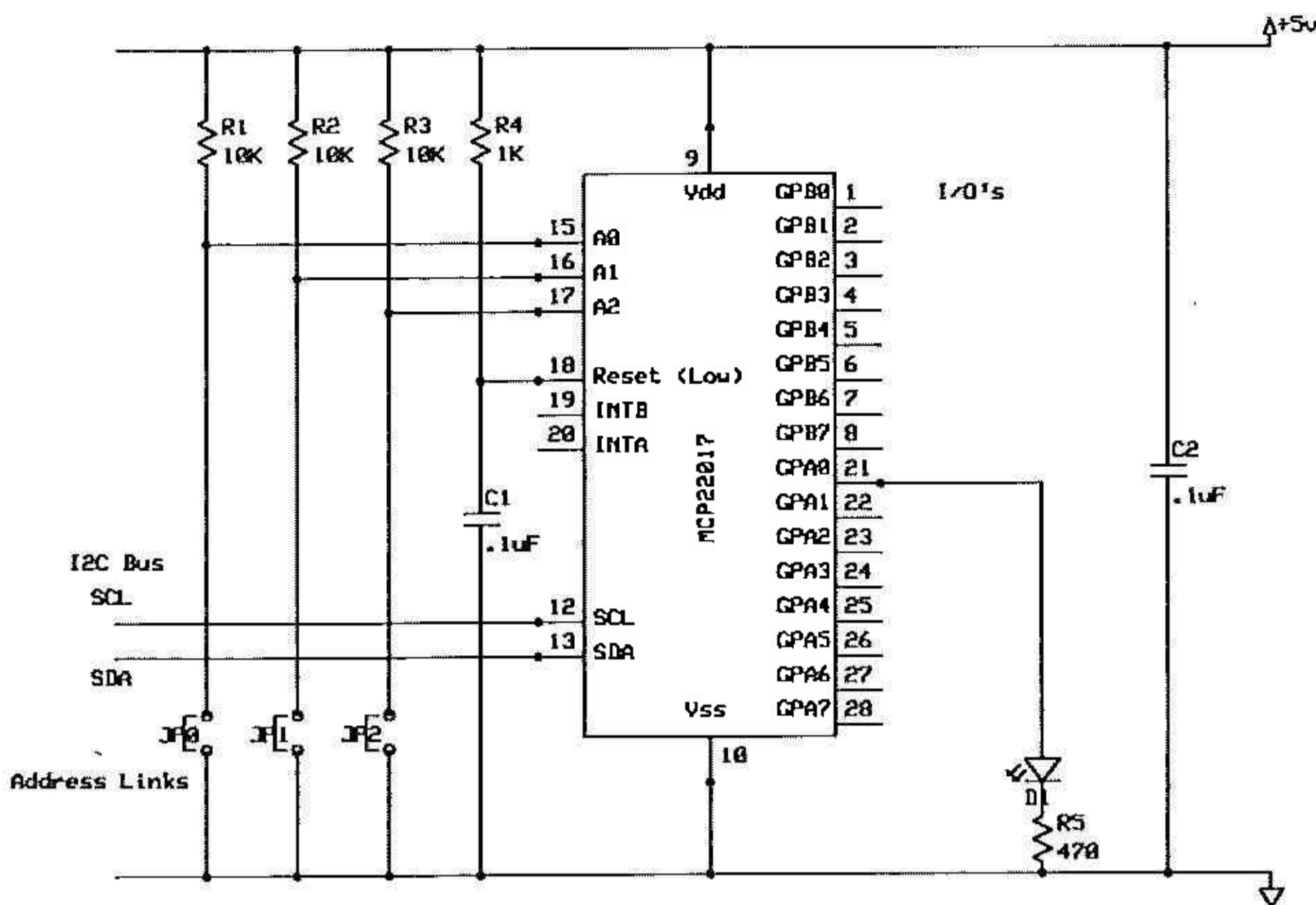
So, that's the end of the second exciting instalment of LibGen. Next time, we will add the real meat of the utility - the ability to load a file, select which entries you want from a menu, and then save those entries out to a file suitable for inclusion in future assembly programs.

I2C Interface for QL Emulators

Part 6

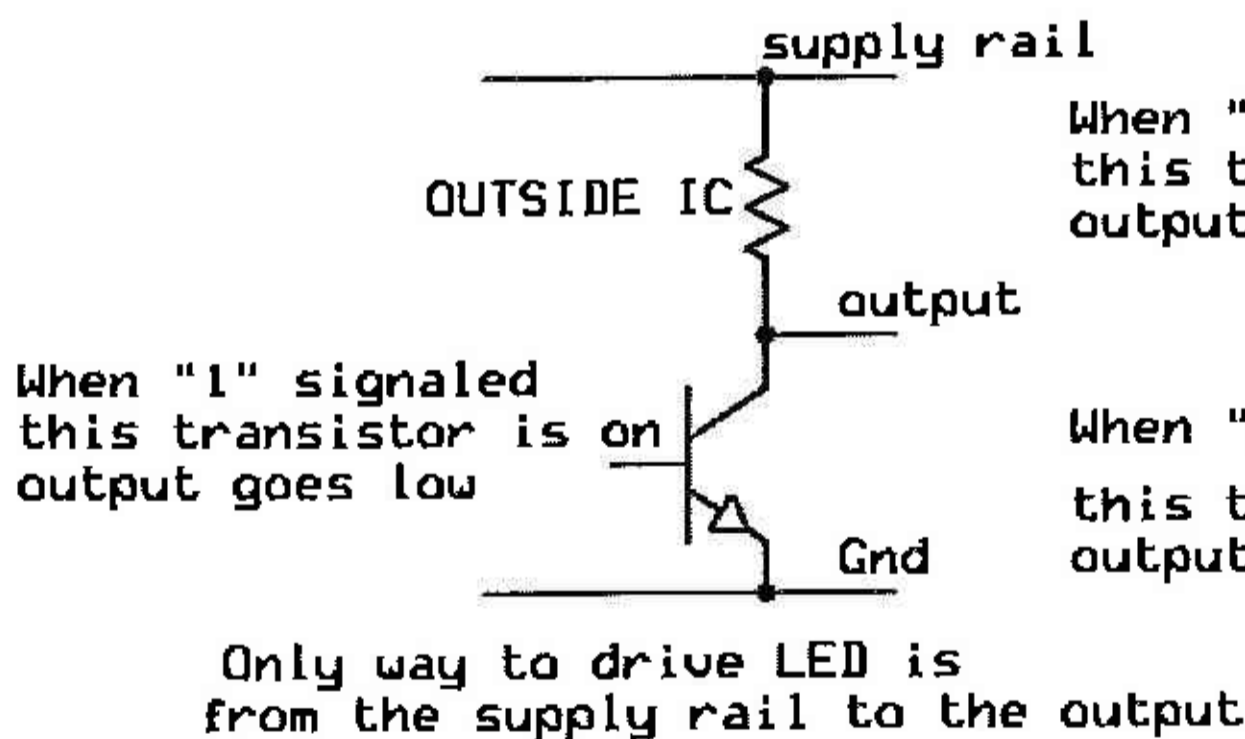
by Ian Burkinshaw

I recently purchased some PCF8574A's and found the price was now fairly high at £4.68 plus VAT each from Farnell and £1.63 plus VAT at RS Components. So I looked to see if there were any other devices about, and came across a device from Microchip of PIC fame. Called MCP23017 (Farnell part number 1332088) at 95p plus VAT or from RS Components (part number 403-806) for 78p plus VAT. Please be aware there may be a minimum order from these companies, so it is worth checking out other suppliers. It is even better value than the PCF devices as it has 2 * 8 bit ports, so 16 GPI (general purpose interface) lines, which otherwise would need 2 of the PCF devices - a major saving either way.

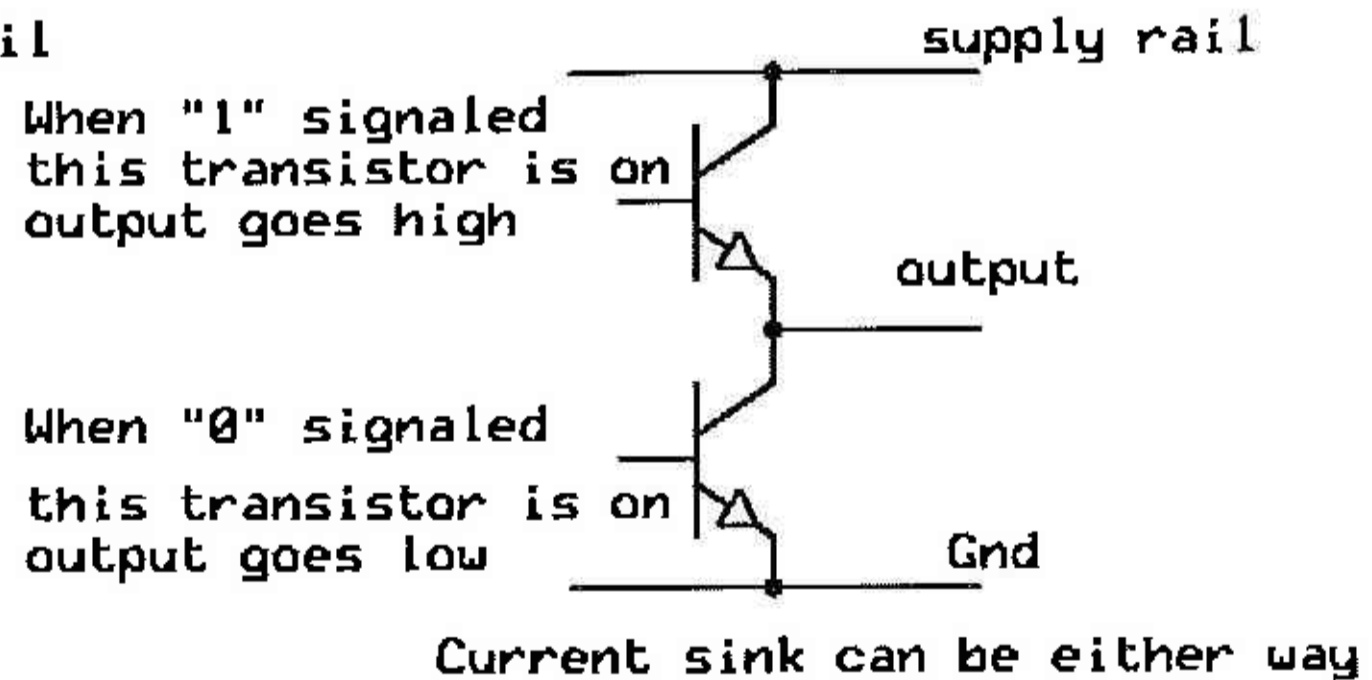


As you can see the circuit is straightforward, the main difference, apart from the 16 I/O's is the reset pin (18) – this must be held high. As you may have seen in the address table in the part 5 of this series, the MCP23017 has the same address range as the PCF8574, so 8 of these device can be used on each I2C bus giving a maximum of 128 I/O lines. Put another way, it is the same as using 8 * PCF8574 and 8 * PCF8574A with a lot less wiring to do.

Another major hardware difference is the way the outputs work. With both the PCF8574 and PCF8574A, when in output mode, the output is what is called sink only. That is, there is, in effect, a transistor between the output pin and the GND, with a weak resistor from the output pin to the 5V supply rail, which is why when a given pin is given a 'high', it is in fact low. Or put another way, it is inverted. This is one of the reasons, I put a driver chip ULN2803 to drive the LED's on my test card (see part 1 of this series) to make the LED's display in the correct sense. In the case of the MCP23017, it has a more normal TTL type output. Sometimes referred to as a totem pole output.



PCF8574(A) Output arrangement



MCP23017 Output arrangement

This is diagrammatic, this is not how it is really done within the chip. This is just for explanation.

There are, in effect, two transistors, one from the output pin to GND, as in the PCF example, but the resistor in the PCF is now replaced with another transistor. One transistor is turned on at a time. The output can sink from supply to GND or to +5. For example you can connect LED's from the output pin (LED anode), to GND (LED cathode) via a current limiting resistor with no extra chip to invert the sense. Another cost saving. However, you should not draw more than 20mA from any pin. If you need more current then you will still need something like the ULN2803 chip or drive power transistors. I shall leave this to you since it will depend on your application. In addition the BV4221 can only provide up to a maximum of 100mA. If your application requires more current, then you will have to provide an external power supply.

QUANTA

QL USERS AND TINKERERS ASSOCIATION

World-wide membership is only by subscription and offers the following benefits:

Bimonthly Magazine - up to 56 pages

Massive Software Library - All Free! : Free Helpline and Workshops

Regional Sub-Groups: Is there one near you?

Free advice on Software and Hardware problems

1 year Membership Subscription £18 (includes eMag)

**If you want a printed copy of QUANTA magazine,
add the following postage rates**

The 2013 postage rates are: UK £4.00, Europe £22.50 and ROW £29.50

PayPal Surcharge about 5% - PayPal (see QUANTA Web Site)

Cash, Cheques and Postal Orders Accepted

***** Now in our Twenty Ninth Year *****

Email: membership@quanta.org.uk

Website: <http://www.quanta.org.uk>

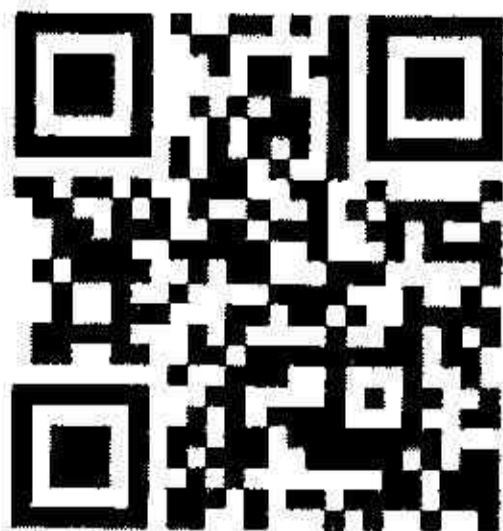
Further details from the Membership Secretary

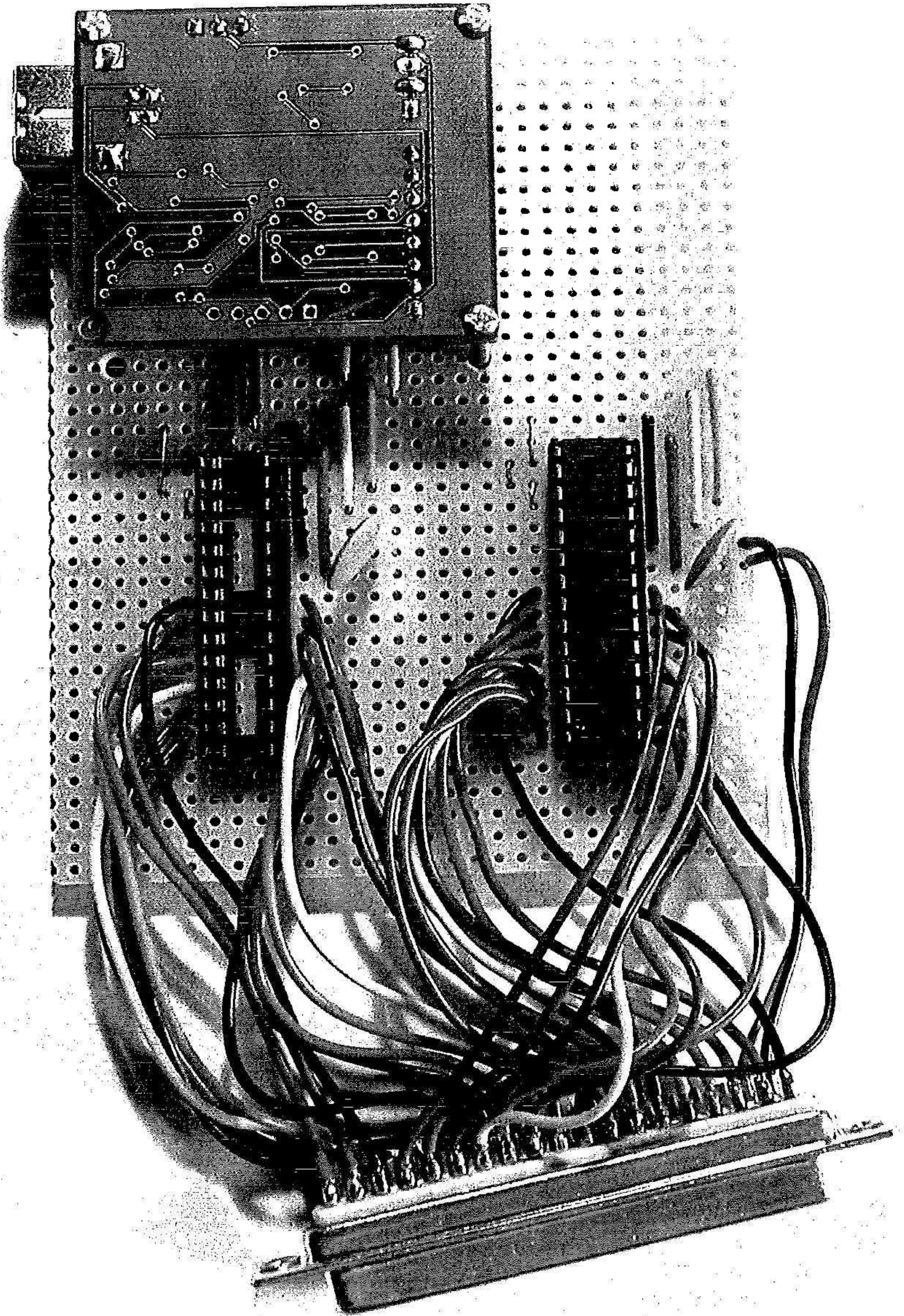
John Gilpin, 181, Urmston Lane, Stretford

Manchester. M32 9EH (UK)

Tel. 0161 865 2872

Ask for our special 3 Year discount





The picture above is my test circuit with two MCP23017 IC sockets with a BV4221-V2.

Now the software side of things

The MCP23017 has 22 registers, do please look at the datasheet for the functionality of these registers. However I shall show in the following routines how to toggle the outputs for both 'A' and 'B' ports. The important thing to bear in mind is you have to send the start signal, then the device address, then the register address, the value for that register, and finally the stop signal. So for example to set the 'A' port to output mode we send 's-40 00 00 p'. Then we can send 's-40 14 ff p', this will then set all the outputs on port 'A' high. 40 is the address 'in hex' of the device if all the address links are in place, i.e. all the address pins are low. Also note all hex letters must be lower case.

I shall not repeat the common routines you need to make all this work. See QLT Vol 16 issue 1. You need lines 1000 to 3090 from this listing.

Until next time, when I will be looking at a RS232 to I2C converter you can use with a 'Black Box' QL.

References

http://www.byvac.com/bv3/index.php?route=product/product&product_id=88

(Please note, I used the original V1 of the BV4221. ByVac now supply V2 which also has a SPI interface. The commands are the same, so the programs listed in the article should still work.)

<http://www.byvac.com/bv3/index.php?route=product/category&path=44>

PCF8570 Ram Data Sheet

http://www.nxp.com/documents/data_sheet/PCF8570.pdf

PCF8574(A) Data Sheet

http://www.nxp.com/documents/data_sheet/PCF8574.pdf

<http://focus.ti.com/lit/ds/symlink/pcf8574.pdf>

PCF8591 Data Sheet

http://www.nxp.com/documents/data_sheet/PCF8591.pdf

DS1307 RTC (Real Time Clock)

<http://datasheets.maxim-ic.com/en/ds/DS1307.pdf>

DS1803 Digital Potentiometer Data Sheet

<http://datasheets.maxim-ic.com/en/ds/DS1803.pdf>

MCP23017 Data Sheet

<http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en023499>

I2C Tutorials

http://www.robot-electronics.co.uk/acatalog/I2C_Tutorial.html

http://www.i2c.byvac.com/ar_foundation.php

http://doc.byvac.com/index.php5?title=I2C_Foundation#SPI

TF Services I2C manual

<http://www.dilwyn.me.uk/docs/manuals/index.html>

Advanced I2C information, but still worth a read to understand I2C protocols

Glossary of Abbreviations and Terms

Part 6 - P to Q

by Dilwyn Jones
and Lee Privett

- PAL** Programmable Array Logic, a type of logic chip, also used for Phase Alternate Line - most of the world uses this television broadcast encoding system
- Pandora** Name given to a case system made by members of the NEMQLUG (North East Manchester) QL user group. This case was specially made to allow you to fit a QL or Aurora circuit board into a neat and portable metal black case.
- PAR** An abbreviation used to represent a parallel printer port, sometimes known as a Centronics compatible printer port. On an original QL, you'd tell programs to print to SER1 or SER2. If you use the printer port on a Super Gold Card, or on a PC fitted with a QXL or QPC emulator, it will usually have the name PAR as far as the QL system is concerned, so you'd tell programs to print to PAR instead of SER1, for example. A parallel port differs from a serial port in that it can send several bits of information down a cable at the same time (usually 8) rather than 1 at a time as

would be the case with a serial port. Printing via a parallel port would usually be faster than to a serial port, but the disadvantages could be that (a) cables have to be shorter than serial links for reliability, and (b) information can usually only go out from the computer to whatever is connected, you cannot get much information back, so you couldn't connect two computers together to share information via the parallel port on a Super Gold Card, for example.

- Parameters** Values passed to a procedure or function. For example, a function which adds two numbers together would have two parameters, one for each of the two numbers to be added together. For example, DEF FN Add(number1,number2) would be called using two numbers in place of the names "number1" and "number2": LET result=Add(2,3)
- Parser** A part of a compiler or other program (e.g. a program which runs a text adventure game) which processes the text to extract information. This checks that the syntax of the text is correct and as it should be for the application. For example, the Turbo compiler parses a SuperBASIC or SBASIC program to ensure that the text of the program is correct before it tries to compile the program.
- Pass by value or reference** Refers to the method by which parameters are passed to a procedure or function. In effect, it refers to whether or not the value of the variable passed to the routine is changed upon return from the routine. In effect, 'pass by value' means that a copy of the value is passed to the routine and even if you try to change its value within the procedure or function, the original value is restored when you exit. 'Pass by reference' means that if you change the value of the variable within the routine, that affects the variable itself, not the copy, so the change remains when you exit from the routine.
- PCB** Printed Circuit Board, electronic boards where components are placed to make circuits, normally manufactured
- PCL** Printer Control Language, a Hewlett Packard system for controlling printers. There are various levels of this, up to level 5, and commonly used with laser printers. Also used to some degree by Deskjet and similar inkjet printers
- PD** Public Domain
- PE** Abbreviation for Pointer Environment. See PTR_GEN. PE normally refers to just the facilities provided by PTR_GEN alone, whereas the term Extended Environment (EE) refers to the combined facilities of the files PTR_GEN, WMAN and HOT_REXT.
- PIPE** A kind of conduit used to send data from one place to another on the QL, analogous to a pipe used to carry water from one place to another, for example.
- Pixel** A Picture Element, one dot of a screen picture or graphic. Pictures are generally referred to as having so many pixels across and so many down. A mode 4 screen from an original Sinclair QL would have 512 pixels across and 256 pixels down, making a total of 131,072 pixels on the screen.
- PNG** The letters stand for Portable Network Graphics. This is a graphics compression system created in the 1990s.
- Pointer Environment** see PE above.
- Port** 1. to rewrite a program to work on a different processor or operating system, or 2. an electrically wired I/O system in a processor used for communication between the processor and other devices or other computers.

Procedure	A block of code, usually given a name, which is used to perform one or more specific tasks within a program. Think of it as a building block for a program. It can be called from other places in a program several times by different parts of a program, to avoid having to rewrite the same code over and over again. Unlike a function (q.v.) a procedure does not return a value upon terminating.
Proforma	Name of software from PROGS (q.v.) which provided a system whereby other specially written software could create Vector Text (q.v.) and Vector Graphics.
Program	We tend to use the word Program without thinking about how to define it. In simple terms, it is a set of instructions for a computer to perform a specific task.
PROGS	Belgian software house run by Joachim van Der Auwera. Produced a wide range of QL software such as The Painter, Line Design, Proforma and ProWeSS.
ProWeSS	The PROGS Windowing sub-system. Advanced Window Manager software from QL software house PROGS.
PSU	Power Supply Unit
PTR_GEN	The pointer interface for the QL's windowing system. The file PTR_GEN (or its equivalent in SMSQ/E) is responsible for controlling a mouse pointer on the screen and for saving and restoring the contents of program windows as you switch between programs with CTRL-C (see above). PTR_GEN is supplied with most pointer environment programs, such as QPAC2.
Q+4	A large QL expansion unit sold by CST in the 1980s. It allowed up to four expansion cards to be added to the QL at a time. It consisted of a large metal base unit on which the QL stood and into which up to 4 cards could be plugged. It then had a short ribbon cable which connected the base unit with a small circuit board which plugged into the QL's expansion socket. The Q+4 is now a very rare device.
Q-emuLator	A QL emulator program written by Daniele Terdina. There are versions for Windows, Mac and OSX operating systems, allowing you to run QL software on those systems.
Q40, Q60	Two computers designed by Peter Graf in Germany. These computers can use either the SMSQ/E or QDOS Classic operating systems. They feature displays up to 1024x512 pixels in size and use 16 bit colour
QDOS	QL Drive Operating System or QL Disk Operating System. This is the operating system of the QL, which is basically what makes it tick. QDOS is responsible for starting up the QL when you switch it on, and provides the necessary code and routines to let you do anything from printing to the screen to multi-tasking your programs. Tony Tebby (its designer) has also been known to refer to QDOS as Domes-DOS.
QDOS CLASSIC	Name of a version of the QL operating system which runs on (a) the Amiga QL emulators and (b) the Q40 and Q60
QeM	A little known emulator of the QL in software on Atari computers.
QJUMP	A QL company set up by Tony Tebby after he finished working for Sinclair. Tony Tebby designed the QL's operating system and went on to produce major software for the QL such as QRAM and QPAC2. The name lives on in terminology such as "Qjump Standard Configuration Block" (a software system allowing for standardised configuration of QL software).

QLIB	Abbreviation used for the Q-Liberator BASIC compiler from Liberation Software.
QL Today	A magazine about the QL, first produced in 1996. The magazine is published in Germany by Jochen Merz Software.
QL User	A QL magazine from the 1980s. Eventually merged with another magazine called QL World.
QL World	A QL magazine first published in the 1980s. For many years it was the main QL magazine, and ownership changed many times until the magazine was finally closed by publishers Arcwind.
QLAY	A freeware QL emulator program for Windows 95, DOS and Linux based systems. Allows these machines to run QL software, and can be downloaded free from the Web site http://www.inter.nl.net/hcc/A.Jaw.Venema
QL2K	A further development of the Qlay QL emulator by Jimmy Montesinos
QLSSS	See SSS
QMenu	See Menu Extension above.
QPAC	QL Pointer Accessories. Either of two packages produced by Tony Tebby to enhance what you can do with your QL. QPAC gave you a number of small but useful programs such as a calculator and typewriter and alarm clock, while QPAC2 gives you a file handling menu, buttons and all sorts of utilities to help you with the multi-tasking and windowing system on the QL
QPC	QL on PC, a commercial program which allows a PC to run QL software by making the PC pretend to be a QL as far as the software is concerned.
QPTR	People often use this term to mean the Pointer Environment or Extended Environment. However, QPTR is actually the name for a programming toolkit for the pointer environment by Tony Tebby and his old company Qjump. It consisted of a huge manual, plus a floppy disk which had the qptr toolkit and various example files.
GRAM	Predecessor to QPAC2, a collection of menus and utilities vaguely along the lines of QPAC2. GRAM is no longer available
QSAVE	See SAV below.
QTYP	QL Typing checker, from Tony Tebby
QUANTA	The main QL user group, originally set up in 1984, when it was originally called IQLUG (the Independent QL Users Group). After deciding that the name IQLUG wasn't liked by everyone (some found it hard to pronounce) the group's name was changed to QUANTA, which was the name of the group's newsletter. After a while, someone came up with a slightly contrived 'QL Users And Tinkerers Association' as a proposed meaning for the acronym QUANTA. It stuck.
QUBIDE	A hard disk interface for the QL.
QVME	A graphics card available for the Atari ST QL emulator. VME stands for Versa Module Europe, enabling a card size called Eurocard to be used on 68000 based computer systems such as Atari's
QXL	A card which plugs into an ISA (Industry Standard Architecture) slot on a PC, allowing it to run QL software much faster than an original QL. I have no idea what the X stands for – probably implying extended QL or something like that.

Although the postage situation for next year is not looking good for us, we have to find a solution somehow. There is a possibility that the next issue will be shipped from Austria ... but there is no solution for the summer issue yet.

Apart from this, this year's QL Today issues were very enjoyable - thanks to you, authors and subscribers! Loads of material - very interesting stuff, many more "hardware" articles than I ever expected... I am glad the QL scene is still as active and alive as it is and I really look forward to see some/many(?) of you in Hamburg next year!

HAPPY XMAS

&

HAPPY NEW YEAR

**FROM THE
QL TODAY
TEAM**

The QL Show Agenda

Very good News - QL Show in Germany in 2013 - in Hamburg!

We need your vote to find the best date!

It seems that the article in the last issue of QL Today has triggered something for next year.

The venue this time will be Hamburg ... a nice city and worth visiting, if you have not been there before. And if you have, worth another visit too... as the address will be "Reeperbahn 1" (famous!) - the entry to St. Pauli.

Rainer Wolkwitz offered conference rooms for the meeting for up to 200 people in den "Tanzende Türmen"/STRABAG ... and he says it will have a beautiful view over Hamburg and its port (only 10 minutes walking distance from there). The towers are quite interesting to look at, especially if you stay in front of them. You can search the web for "Tanzende Türme" to get some impressions about the interesting building.

We thought that the date for the meeting should avoid the holiday season and major events in Hamburg, to increase chances for reasonable hotel rates.

We found that the following dates are probably the best ones, chances for good weather, chances for less traffic, chances for affordable hotels:

- a) Weekend 18th/19th of May
- b) Weekend 25th/26th of May
- c) Weekend 1st/2nd of June
- d) Weekend 31st of August/1st of September

So, we should start working out which date suites most of you best, and if you prefer the Saturday or the Sunday to be the "meeting" date.

Assuming, that a visit to Hamburg will not just be for a QL meeting, other activities like siteseeing or shopping can easily be placed around the meeting. A note for visitors from outside Germany: shops are closed on Sunday, but there are loads of other things to do, e.g. like visiting the harbour and various ships, or cruise through the harbour, musicals, Zoo Hagenbeck, all sorts of museums or the famous Miniatur Wunderland (the world's largest miniature train exhibition).

Please let us know if there is any interest, and what would be your preferred date and day ... please send an email to SMSQ@J-M-S.com, subject "QL2013" and the letters for your TWO preferred dates (a,b,c,d) and whether you prefer Saturday or Sunday, or no preference.

Also, help would be most welcome - Rainer said he does not have too much time to care for other things like hotels. He can provide technical equipment which could be required for talks or demonstrations (hint, hint!), like a large screen or projector and internet, of course.

Please reply a.s.a.p. so that we can start working on the details. For dates in May, only half a year is left, which is 2 issues of QL Today in between a poll of the preferred date and advertising the event. We may not be able to fill the room with 200 people, but after having had several meetings in the South, this is very North and a chance to meet people we have not seen for many years!

And - Hamburg is worth visiting and seeing anyway!

